

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

**RESTAURACE SIGNÁLU S OMEZENOU OKAMŽITOU
HODNOTOU S POUŽITÍM PSYCHOAKUSTICKÉHO
MODELU**

RESTORATION OF SIGNALS WITH LIMITED INSTANTANEOUS VALUE USING A PSYCHOACOUSTIC
MODEL

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Tomáš Beňo

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Pavel Závíška

BRNO 2019

Diplomová práce

magisterský navazující studijní obor **Audio inženýrství**

Ústav telekomunikací

Student: Bc. Tomáš Beňo

ID: 174443

Ročník: 2

Akademický rok: 2018/19

NÁZEV TÉMATU:

Restaurace signálu s omezenou okamžitou hodnotou s použitím psychoakustického modelu

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte znehodnocení zvukového signálu s omezenou okamžitou hodnotou, tedy jev nazývaný též jako clipping. Problém analyzujte a seznamte se s obvyklými metodami rekonstrukce takovýchto signálů. Prostudujte především metody založené na tzv. řídkých reprezentacích signálů.

Následně v prostředí MATLAB implementujte rekonstrukční algoritmus pro "declipping" na bázi řídkých reprezentací signálů. Nastudujte také psychoakustický model používaný v kompresním standardu MPEG, který následně implementujte a rekonstrukční algoritmus jím rozšiřte.

Proveďte důkladnou analýzu nastavení různých parametrů algoritmu a použití psychoakustického modelu za účelem dosažení co nejlepší kvality rekonstrukce. Výsledky rekonstrukce pak vyhodnoťte pomocí vhodných objektivních i subjektivních ukazatelů.

DOPORUČENÁ LITERATURA:

[1] Hrbáček, R., Rajmic, P., Veselý, V., Špiřík, J. Řídké reprezentace signálů: úvod do problematiky, Elektrevue, 2011. ISSN 1213-1539.

[2] Defraene, B.; Mansour, N.; De Hertogh, S.; Van Waterschoot, T.; Diehl, M.; Moonen, M. Declipping of Audio Signals Using Perceptual Compressed Sensing. IEEE Transactions on Audio, Speech, and Language Processing [online]. 2016. s 2627–2637.

Termín zadání: 1.2.2019

Termín odevzdání: 16.5.2019

Vedoucí práce: Ing. Pavel Záviška

Konzultant:

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Diplomová práce se zabývá restaurováním zvukových signálů, které byly poškozeny clippingem. Mezi zde využitými metodami patří ty, které jsou založeny na řídkých reprezentacích signálů. V úvodu práce je vysvětlena problematika clippingu a zmíněn výčet již existujících metod, které řeší declipping, a na které tato práce navazuje. V další části je popsána nezbytná teorie týkající se řídkých reprezentací a teorie týkající se proximálních algoritmů, včetně konkrétních zástupců z této kategorie úloh řešících konvexní optimalizaci. Součástí této práce je algoritmus pro declipping implementovaný v prostředí Matlab. Metoda zvolená pro řešení této úlohy využívá Condatův algoritmus neboli Obecný proximální algoritmus pro konvexní optimalizaci a řeší minimalizaci součtu tří konvexních funkcí. Výsledkem je pět odlišných variant algoritmu, z nichž tři mají implementovaný také psychoakustický model za účelem dosažení lepších výsledků. Pro každou variantu bylo nalezeno optimální nastavení parametrů. Výsledky rekonstrukce jsou vyhodnoceny pomocí objektivních ukazatelů Δ SDR a PEMO-Q a také pomocí subjektivního poslechového testu.

KLÍČOVÁ SLOVA

restaurování signálu, clipping, declipping, řídké reprezentace signálů, l1-optimalizace, proximální algoritmus, Condatův algoritmus, psychoakustický model, MPEG, globální maskovací práh

ABSTRACT

The master's thesis deals with the restoration of audio signals that have been damaged by clipping. Used methods are based on sparse representations of signals. The introduction of the thesis explains the issue of clipping and mentions the list of already existing methods that solve declipping, which are followed by the thesis. In the next chapter, the necessary theory of sparse representations and the proximal algorithms is described, including specific representatives from the category of convex optimization problems. The thesis contains declipping algorithm implemented in Matlab software environment. Chosen method for solving the task uses the Condat algorithm or Generic proximal algorithm for convex optimization and solves minimization of sum of three convex functions. The result of the thesis is five versions of algorithm and three of them have implemented psychoacoustic model for results improvement. For each version has been found optimal setting of parameters. The restoration quality results are evaluated using objective measurements like Δ SDR and PEMO-Q and also using subjective listening test.

KEYWORDS

signal restoration, clipping, declipping, sparse representations, l1-optimization, proximal algorithm, Condat algorithm, psychoacoustic model, MPEG, global masking threshold

BEŇO, Tomáš. *Restaurování signálu s omezenou okamžitou hodnotou s použitím psychoakustického modelu*. Brno, 2019, 85 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Pavel Závíška

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Restaurace signálu s omezenou okamžitou hodnotou s použitím psychoakustického modelu“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Pavlu Záviškovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci. Také tímto děkuji všem respondentům, kteří se ochotně zúčastnili poslechového testu.

Brno

.....

podpis autora

Obsah

Úvod	11
1 Clipping zvukového signálu	12
1.1 Obecné poznatky	12
1.2 Hard clipping	13
1.3 Soft clipping	13
1.4 Vlastnosti clippingu	14
1.5 Clipping v hudbě	17
1.6 Další typy poškození signálu	17
2 Metody restaurace zvukových signálů	18
2.1 Declipping	18
2.2 Metody declippingu využívající řídké reprezentace signálů	19
2.2.1 A Constrained Matching Pursuit to Audio Declipping	19
2.2.2 Consistent Iterative Hard Thresholding for Signal Declipping	19
2.2.3 Declipping of Audio Signals Using Perceptual Compressed Sensing	19
2.2.4 Audio Declipping by Cosparsity Hard Thresholding	20
2.2.5 Sparsity and Cosparsity for Audio Declipping: A Flexible Non-convex Approach	20
2.2.6 Nejnovější příspěvky v oblasti SPADE algoritmů	21
3 Psychoakustika	22
3.1 Maskování	22
3.1.1 Práh slyšení	22
3.1.2 Maskování v kmitočtové oblasti	23
3.1.3 Maskování v časové oblasti	24
4 Hodnocení kvality restaurace signálů	26
4.1 Objektivní metody	26
4.1.1 SDR	26
4.1.2 PEMO-Q	27
4.2 Subjektivní metody	28
4.2.1 MUSHRA	28
5 Řídké reprezentace signálů	30
5.1 Výchozí základy a značení	30
5.2 Normy vektorů	30
5.3 Řídkost vektoru	31

5.4	Řídká řešení systémů lineárních rovnic	31
5.4.1	Postačující podmínky jednoznačného řešení	32
5.5	ℓ_1 -relaxace	33
6	Báze a frame	35
6.1	Generátory vektorového prostoru	35
6.2	Báze vektorového prostoru	35
6.3	Frame	36
6.4	Gaborův Frame	37
7	Proximální algoritmy	38
7.1	Vymezení proximálních algoritmů	38
7.2	Proximální operátory	38
7.3	Metody proximálního dělení	39
7.4	Condatův algoritmus	40
7.5	Douglas-Rachfordův algoritmus	42
7.6	Chambolle-Pockův algoritmus	43
8	Řešení úlohy declippingu	44
8.1	Obecná formulace	44
8.2	Konkrétní řešení pomocí Condatova algoritmu	45
8.2.1	Formulace Condatova algoritmu	45
8.2.2	Nastavení parametrů	46
8.2.3	Algoritmus a jeho popis	47
8.3	Řešení s využitím psychoakustického modelu	47
8.3.1	Perceptuální váhovací funkce	48
9	Programové řešení úlohy	51
9.1	Strukturace kódu	51
9.2	Hlavní soubor 1	51
9.2.1	Parametry	52
9.2.2	Načtení vstupního signálu	52
9.2.3	Clipping	52
9.2.4	Příprava před zpracováním	53
9.2.5	Algoritmus declippingu	53
9.2.6	Výsledky rekonstrukce a jejich zobrazení	53
9.3	Hlavní soubor 2	54
9.3.1	Psychoakustický model	54
9.4	Hlavní soubor 3	55
9.5	Condatův algoritmus	56

9.5.1	Projekce na množinu	56
10	Výsledky testování a měření	57
10.1	Obecné poznatky k testování	57
10.1.1	Testovací zvukové signály	58
10.1.2	Průběžné zobrazení výsledků	58
10.2	Vizualizace výsledků	59
10.3	Vyhodnocení rekonstrukce na základě Δ SDR	60
10.4	Vyhodnocení rekonstrukce na základě PEMO-Q	63
10.5	Srovnání doby rekonstrukce	65
10.6	Poslechový test MUSHRA	66
10.6.1	Metoda testu	66
10.6.2	Aplikace Scale	66
10.6.3	Podmínky a průběh testu	67
10.6.4	Výsledky testu	67
10.7	Porovnání výsledků testu MUSHRA a PEMO-Q	69
10.8	Testování parametrů algoritmu	69
10.8.1	Parametr γ	70
10.8.2	Parametr σ	71
10.8.3	Parametr ρ	71
10.8.4	Váhovací okno	72
10.8.5	Dynamický rozsah	73
10.9	Shrnutí dosažených výsledků	73
11	Závěr	75
	Literatura	77
	Seznam symbolů, veličin a zkratk	83
	Seznam příloh	84
A	Obsah přiloženého CD	85

Seznam obrázků

1.1	Převodní charakteristika hard clippingu.	13
1.2	Převodní charakteristika soft clippingu.	14
1.3	Časová oblast.	15
1.4	Kmitočtová oblast.	15
1.5	Spektrogramy zvukového signálu.	16
3.1	Závislost prahu slyšení na kmitočtu.	23
3.2	Maskovací práh harmonického signálu o kmitočtu 1 kHz a hladině 94 dB(SPL).	24
4.1	Blokové schéma metody PEMO-Q.	28
5.1	Zobrazení jednotkové koule pro různé normy.	34
7.1	Znázornění projekce vektoru \mathbf{x} na množinu C	39
7.2	Blokové schéma declippingu pomocí Condatova algoritmu.	42
8.1	Blokové schéma declippingu pomocí Condatova algoritmu s využitím psychoakustického modelu.	50
10.1	Zobrazení typických průběhů při monitorování chování algoritmu. . .	58
10.2	Použité nástroje pro vizualizaci výsledků.	60
10.3	Závislost ΔSDR na úrovni θ_c	61
10.4	Závislost ΔSDR na úrovni θ_c pro jednotlivé metody.	62
10.5	Závislost hodnoty na stupnici ODG na úrovni θ_c	63
10.6	Závislost ODG na úrovni θ_c pro jednotlivé metody.	64
10.7	Závislost počtu iterací na úrovni θ_c pro jednotlivé metody.	65
10.8	Výsledek testu MUSHRA pro jednotlivé metody.	68
10.9	Porovnání výsledku PEMO-Q a testu MUSHRA pro jednotlivé metody. .	69
10.10	Závislost ΔSDR na velikosti parametru γ	70
10.11	Závislost ΔSDR na velikosti parametru σ	71
10.12	Závislost ΔSDR a doby rekonstrukce na velikosti parametru ρ	72
10.13	Vliv vlastností váhovacího okna na ΔSDR	72
10.14	Závislost ΔSDR na velikosti θ_c pro různé nastavení dynamického rozsahu.	73

Seznam tabulek

4.1	Pětibodová škála hodnocení poškození zvukových signálů ODG	27
4.2	Stupnice hodnocení metody MUSHRA	29
10.1	Číselné označení použitých testovacích hudebních ukázek.	68

Úvod

Poškození zvukových signálů je v praxi častým jevem, ke kterému dochází z různých příčin. Jedním z těchto typů poškození je tzv. *clipping*, který je na poslech velmi nepříjemný a navíc může poškodit prvky elektroakustického řetězce. Pokud se nepodaří zajistit dostatečnou prevenci proti vzniku clippingu, přichází na řadu způsoby, jak takto poškozený signál následně opravit a přiblížit jej co nejvíce originálu. Samotný termín clipping je nyní už běžně v této podobě přejímán do češtiny, ale při důsledném označení se jedná o omezení okamžité hodnoty signálu.

První pokusy o *declipping*, což je operace, jejímž úkolem je minimalizovat škody způsobené clippingem, pochází již z roku 1991 [1]. Od té doby tato oblast prošla značným vývojem a v současnosti se používají mnohem pokročilejší matematické metody, které jsou realizovatelné i díky značnému výkonu současné výpočetní techniky, ve srovnání s 90. léty minulého století. V této práci je pro *declipping* použita metoda založená na řídké reprezentaci, což je v současnosti poměrně progresivní oblast zpracování signálů.

První kapitola obsahuje seznámení se samotnou problematikou clippingu, kde jsou rozebrány charakteristiky a příčiny vzniku tohoto jevu. V kapitole 2 je kromě popisu operace *declippingu* také seznam a stručný popis prací, které se v minulosti touto problematikou zabývaly. Tato práce na ně prakticky navazuje a hledá možnosti využití psychoakustického modelu pro dosažení dalších zlepšení v této oblasti.

Kapitola číslo 3 obsahuje teoretický základ z oblasti psychoakustiky, zejména oblasti, které budou využity v této práci. Další kapitola se týká hodnocení kvality restaurace signálů, kde jsou popsány způsoby, jakými se vyhodnocuje kvalita rekonstrukce poškozených zvukových signálů. Následující kapitoly 5 a 6 seznamují s nezbytnou teorií řídkých reprezentací signálů, která je podstatná pro pochopení principů, na kterých jsou založeny algoritmy pro *declipping*. Ty jsou rozebrány v kapitole 7.

V kapitole 8 je formulována nejprve obecná úloha *declippingu* a následně konkrétní řešení pomocí Condatova algoritmu. Na konci kapitoly je popsáno řešení s využitím psychoakustického modelu. Programové řešení úlohy obsahující popis kódu programu je náplní kapitoly 9. Poslední kapitola se věnuje rozboru dosažených výsledků a pomocí nástrojů popsaných v kapitole Hodnocení kvality restaurace signálů jsou interpretovány výsledky objektivních testů pomocí ukazatele ΔSDR a PEMO-Q a subjektivního poslechového testu.

1 Clipping zvukového signálu

Tato kapitola popisuje jev zvaný clipping a jeho vlastnosti. Dále je zde popsáno a vysvětleno rozdělení na *hard clipping* a *soft clipping*, jejich vznik a charakteristický projev. V části 1.5 je nastíněno umělecké využití clippingu v oblasti hudby, protože se nemusí jednat o negativní jev za všech okolností. Na konci kapitoly je krátká zmínka o dalších možných typech poškození zvukového signálu.

1.1 Obecné poznatky

Clipping neboli omezení okamžité hodnoty signálu je druh nelineárního zkreslení, který postihuje signály v analogové i číslicové oblasti při překročení dynamického rozsahu systému. V oblasti zpracování signálů může být clipping důsledkem převodu z analogové do digitální oblasti a naopak, či obecně kdykoliv během transformace, kde figuruje změna zesílení. V číslicové oblasti k němu dojde, když je signál omezen rozsahem počtu bitů dané číselné reprezentace [2]. V praxi se lze s clippingem často setkat při nahrávání zvuku, při překročení dynamického rozsahu převodníku. Signál, který se dostane mimo zpracovatelný rozsah se přichytí na horní a dolní prahovou úroveň. Původní hodnoty nad a pod touto úrovní jsou limitovány, resp. „oříznuty“ (z anglického slova *clip*) a dojde tak ke ztrátě informace, která byla původně obsažena ve špičkách signálu [3].

Rozlišují se dvě podoby clippingu. Hard clipping neboli tvrdé ořezání znamená, že je signál striktně omezen na úrovni prahové hodnoty. Lze se setkat i s označením *digitální clipping*. U soft clippingu dochází k postupné komprimaci špiček signálu již před prahovou hodnotou a úroveň komprimace je tím větší, čím více se vstupní signál blíží této hodnotě. Na rozdíl od hard clippingu zde lze při znalosti převodní charakteristiky a přijatelné úrovni oříznutí rekonstruovat zpět původní signál [4].

K soft clippingu dochází mimo jiné při pokusu překročit maximální rozsah výstupního napětí zesilovače. Při překročení této úrovně se zesilovač dostává do stavu tzv. saturace, kdy výstupní napětí odpovídá napájecímu napětí. Omezení může být způsobeno také proudovými a teplotními limity daného zesilovače [5].

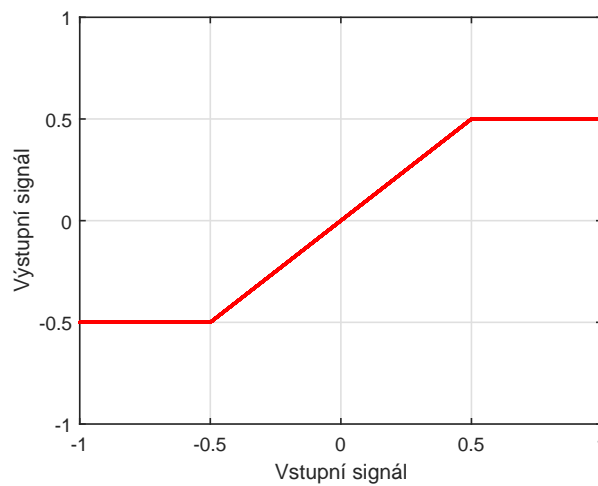
Zpracování zvuku není jedinou oblastí, kde se s clippingem dá setkat. V oblasti zpracování obrazu lze na clipping narazit u fotografií, kde vzniká jako následek překročení dynamického rozsahu světelného snímáče, a lze jej rozpoznat podle bílých nebo černých oblastí na snímku [6].

1.2 Hard clipping

Jak již bylo výše naznačeno, rozlišují se dva druhy clippingu, a to podle tvaru převodní charakteristiky. Hard clipping (případně *clip distortion*) způsobující nekompromisní oříznutí špiček signálu po překročení prahové úrovně θ_c lze vyjádřit jako:

$$y_c[n] = \begin{cases} y[n] & \text{pro } |y[n]| < \theta_c, \\ \theta_c \cdot \text{sgn}(y[n]) & \text{pro } |y[n]| \geq \theta_c, \end{cases} \quad (1.1)$$

kde $y[n]$ figuruje jako vstupní signál a $y_c[n]$ jako výstupní clippovaný signál. Jedná se o symetrickou limitaci, to znamená, že dolní a horní prahová úroveň se liší pouze svým znaménkem.



Obr. 1.1: Převodní charakteristika hard clippingu.

Na základě vztahu (1.1) je vykreslen graf převodní charakteristiky 1.1, zobrazující závislost výstupu na vstupu. Oblast mezi prahovými úrovněmi je lineární a po překročení prahových úrovní je signál omezen na hodnoty odpovídající $\pm\theta_c$, v tomto konkrétním případě $\pm 0,5$.

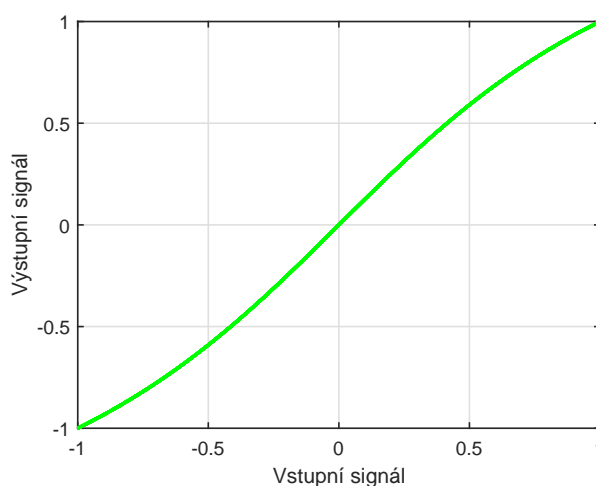
1.3 Soft clipping

Měkkého oříznutí špiček signálu resp. postupné komprese špiček lze obecně dosáhnout modelováním s využitím sigmoidních funkcí jako je například $\tanh(x)$, $\arctan(x)$, $\frac{x}{1+|x|}$ nebo $\frac{x}{\sqrt{1+x^2}}$, případně kubickou nelinearitou uvedenou v [7]. Komplexnější funkce umožňují například i nastavení rozsahu, ve kterém zůstane převodní charakteristika lineární.

V tomto konkrétním případě byla pro soft clipping zvolena poměrně triviální funkce využívající průběh cyklometrické funkce arkus tangens s nastavitelnou úrovní saturace vstupního signálu α (v tomto případě zvoleno $\alpha = 2$):

$$y_c[n] = \begin{cases} \frac{2}{\pi} \cdot \arctan(\alpha \cdot y[n]) & \text{pro } |y[n]| \leq 1, \\ 1 \cdot \text{sgn}(y[n]) & \text{pro } |y[n]| > 1. \end{cases} \quad (1.2)$$

Z grafu převodní charakteristiky na obrázku 1.2 je možné vidět hladký průběh průběh funkce a velmi pozvolný přechod z lineární oblasti do oblasti limitace signálu. Lineární oblast má menší rozsah ve srovnání s převodní charakteristikou hard clippingu.

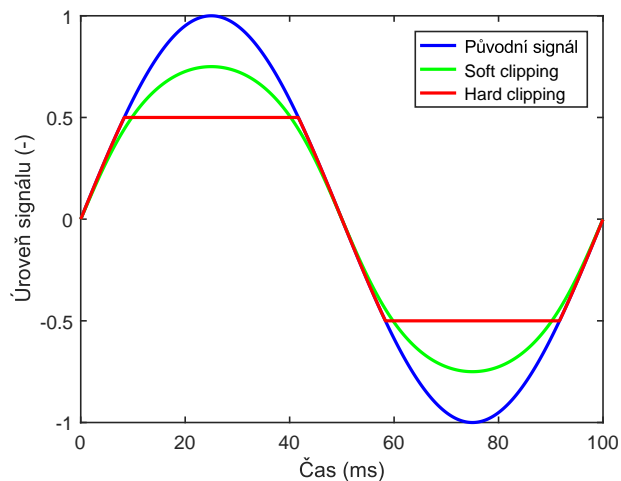


Obr. 1.2: Převodní charakteristika soft clippingu.

1.4 Vlastnosti clippingu

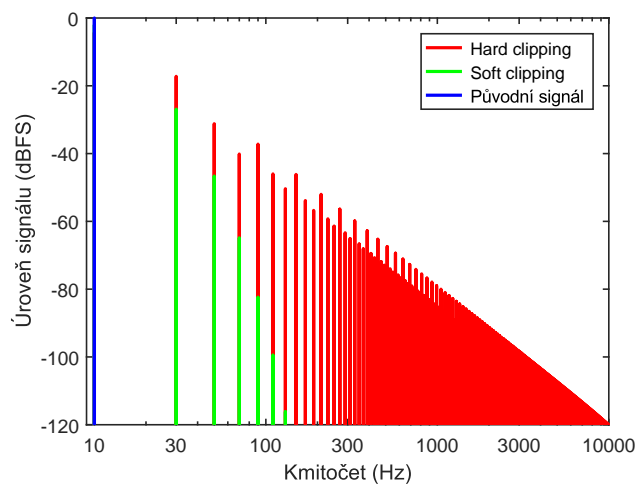
Na clipping, jeho projev a působení na signál je možno pohlížet z více úhlů. Jak lze vidět na obrázku 1.3, v časové oblasti se hard clipping projevuje přichycením časového průběhu na horní nebo dolní mez u vzorků, které překročili prahovou úroveň. Původně sinusový průběh o amplitudě 1 při aplikaci soft clippingu začíná mít čím dál více komprimované špičky, a v případě použití hard clippingu původně sinusový průběh začíná připomínat průběh obdélníkový. Navíc se v průběhu objevují ostré hrany při přechodu k prahovým úrovním.

V kmitočtové oblasti následkem clippingu vznikají nové vyšší harmonické složky signálu. Jejich amplituda a kvantita se odvíjí od tvaru převodní charakteristiky a nastavení prahové úrovně θ_c . Jak lze vidět na obrázku 1.4, při hard clippingu vznikají liché harmonické, jejichž amplituda klesá k vyšším kmitočtům. Výskyt pouze



Obr. 1.3: Časová oblast.

lichých shorků je způsobem tím, že původně hladký průběh funkce sinus nyní obsahuje ve svém průběhu ostré hrany při přechodu tzn. signál už není hladká funkce. Pro výchozí kmitočet 10 Hz tedy vznikly vyšší harmonické na kmitočtu 30 Hz, 50 Hz, 70 Hz, a tak dále.



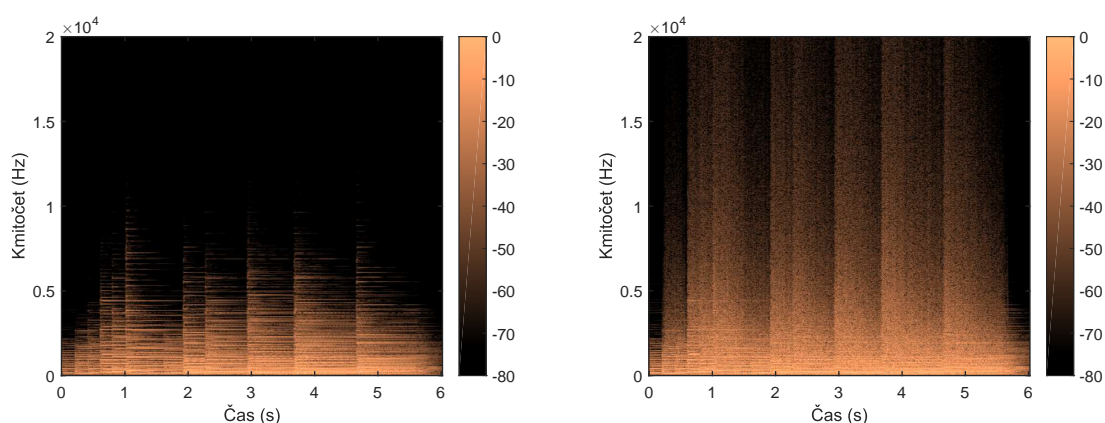
Obr. 1.4: Kmitočtová oblast.

Počet harmonických je teoretický nekonečný, v reálné situaci nastává omezení kvůli vzorkovacímu kmitu. To vede k možnému vzniku aliasingu, tj. zkreslení následkem oboustranného zrcadlení kmitočtového spektra kolem vzorkovacího kmitočtu, a interference zrcadlených složek se signálem ve slyšitelné oblasti [8].

Soft clipping, jehož kmitočtové spektrum je na obrázku 1.4 vykresleno zelenou barvou, má vzhledem k hard clippingu konečný počet nově vzniklých harmonických a jejich amplituda klesá o poznání rychleji. Při použití cyklometrické funkce arkus tangens pro tvarování převodní charakteristiky vznikají opět pouze liché harmonické složky, přestože signál po soft clippingu je stále hladká funkce.

Při popisu zvukové charakteru signálu zasaženého clippingem a zejména hard clippingem lze použít výrazy jako zkreslený, ostrý, průrazný, agresivní či dutý. Tyto pojmy jsou obecně asociované se signály s bohatým kmitočtovým spektrem. S narůstající limitací signálu roste i jeho subjektivní hlasitost při stejné maximální amplitudě. Cenou za vyšší hlasitost je mimo jiné snížení dynamického rozsahu zvukového materiálu. Nově vzniklé harmonické složky v kmitočtovém spektru přinášejí znatelnou změnu barvy, což je subjektivní vjem kvantity, rozmístění a intezity vyšších harmonických [41].

Při pozorování přítomnosti clippingu v audio signálu je velmi praktické zobrazení pomocí spektrogramu, zejména při analýze komplexních zvukových signálů, u kterých informace získané pozorováním časového průběhu ani kmitočtového spektra nejsou dostatečné a mohou vést k chybným závěrům [9]. Spektrogram je dvourozměrný graf, který na svislé ose zobrazuje kmitočet, na vodorovné čas a hladina je vyjádřena barevnou škálou. Tmavá barva reprezentuje nízké hladiny a světlá, měděná barva naopak indikuje vysokou hladinu signálu. Existují ovšem i další používaná barevná schémata. Na obrázcích 1.5(a) a 1.5(b) je vidět porovnání spektrogramu úryvku hry na klavír a jeho silně zkreslené podoby s prahem $\theta_c = 0,1$.



(a) Originální signál.

(b) Silně clippovaný signál.

Obr. 1.5: Spektrogramy zvukového signálu.

1.5 Clipping v hudbě

Obecně je žádoucí se clippingu a jakémukoliv poškození signálu vyvarovat. Na druhou stranu v elektrofonické hudbě si tyto zvukové artefakty našly své místo právě kvůli zmíněným změnám v barvě a vyšší hlasitosti [10]. Zvuk elektronkových zesilovačů pomáhal definovat podobu rockové hudby v druhé polovině minulého století. Zvuk zkreslené elektrofonické kytary se stal brzy fenoménem, který se ještě více rozvíjel s nástupem efektů zkreslujících signál kytar. Jedním z těchto efektů je *overdrive*, který využívá převodní charakteristiku soft clippingu. Efekt *distortion* má výraznější charakter a tzv. *fuzz* má kompletně nelineární převodní charakteristiku a má hrubý a neupravený sonický charakter [11].

Umělecké užití hard clippingu, jehož zvuk je narozdíl od soft clippingu spíše nepříjemný, se poprvé objevilo v okrajových hudebních žánrech počátkem 90. let minulého století. V tu dobu začalo masové používání počítačů pro tvorbu elektronické hudby a záměrný clipping zvukových signálů mohl být realizován mnohem snadněji. Tehdy mladý hudební žánr techno se vyvíjel i do extrémnějších podob, jejichž základem bylo využití hard clippingu, zejména pro syntetické basové bubny [12]. Ještě dál šly subžánry s přídomkem industrial, které jsou charakteristické vysokou mírou zkreslení většiny zvukových prvků [13].

1.6 Další typy poškození signálu

Zvukový signál může být kromě clippingu poškozen i dalšími způsoby. Obecný termín šum lze v tomto kontextu nejlépe definovat jako nežádoucí rušení částečně překrývající původní signál [14]. Šum lze v tomto kontextu rozdělit na náhodný a koherentní. Náhodný šum se obvykle vyskytuje pouze v krátkých časových momentech, jedná se o slyšitelné cvaknutí (z anglického *clicking*) či lupnutí, způsobené krátkými a intenzivními maximy impulzního charakteru. Pokud cvaknutí obsahuje převažující množství nízkých kmitočtů, jedná se o tzv. *pop*, který je vnímán jako tlumené bouchnutí.

Na pomezí mezi koherentním a náhodným šumem se nachází *crackling*. V podstatě se jedná o častý a pravidelný výskyt clicků s nízkou intenzitou, který ve výsledku zní jako praskání (z anglického *crackling*). Typickým příkladem je praskání zaprášených gramofonových desek.

Mezi koherentní šумы patří tzv. *hissing*, co je šum na vysokých kmitočtech projevující se syčivým zvukem a *rumbling*, což je naopak šum na velmi nízkých kmitočtech blízko dolní hranice lidského slyšení [15].

2 Metody restaurace zvukových signálů

Jak bylo zmíněno, již delší dobu existují snahy o restauraci zvukových signálů poškozených clippingem. A to zejména z důvodů zlepšení objektivní i subjektivní kvality zvukového signálu.

V první části této kapitoly je popsán declipping, tedy inverzní operace clippingu, motivace jeho zavedení a také omezující podmínky. V další části jsou stručně rozebrány již existující metody se zaměřením na ty, které pro úlohu declippingu využívají metody řídkých reprezentací signálů.

2.1 Declipping

Declipping je úloha, jejímž cílem je co nejlepší možná rekonstrukce původního signálu, který byl zasažený clippingem [29]. Zjednodušeně řečeno se jedná o doplnění chybějících vzorků, které překročily horní a dolní prahovou úroveň θ_c . Tyto vzorky mají odpovídat těm v původním signálu nebo se jim alespoň přiblížit. Dalším účelem je odstranění co nejvíce pro posluchače rušivých složek, které vznikly jako následek hard clippingu.

Rekonstrukce pomocí declippingu je možná pro soft i hard clipping, ovšem v této práci je pozornost zaměřena na rekonstrukci po hard clippingu s využitím metod řídké reprezentace signálů. Mimoto, pokud u soft clippingu známe převodní charakteristiku a míra zkreslení signálu není vysoká, je možné pomocí inverzní převodní charakteristiky perfektně rekonstruovat původní signál.

Omezující podmínkou pro úspěšné provedení declippingu v rámci této práce je dopředná znalost míst v signálu se znehodnocenými vzorky, se kterou algoritmus pracuje, a také symetrie ořezu hard clippingu.

K této úloze bylo navrženo více přístupů. Prvotní metody byly založeny na provádění interpolace a extrapolace k obnovení clippovaných částí signálu založené na znalostech těch částí signálu, které byly nepoškozené.

Konkrétně se jedná o práci *Restoring a Clipped signal* z roku 1991 autorů Jonathana Abela a Julia Smithe, která je milníkem v této oblasti, protože se jako první přímo věnuje úloze declippingu zvukového signálu [1]. Další přístup spočíval v přijetí vhodného modelu (typicky autoregresivního) čistého signálu a clippingu. Následná obnova probíhala skrze Bayesovský statistický odhad parametrů modelu. Další přístupy k declippingu využívají například neuronové sítě [29].

Lze se setkat i s využitím lineární predikce, jakožto metody analýzy zvukového signálu. Vychází se z předpokladu, že po navzorkování signálu je možné každý další vzorek diskrétního signálu vyjádřit jako lineární kombinaci jeho předchozích vzorků.

Tím pádem každý další vzorek je předvídatelný neboli prediktabilní. Odtud pochází název lineární predikce [16].

2.2 Metody declippingu využívající řídké reprezentace signálů

Mezi další metody, které řeší úlohu declippingu jsou takové, které k tomu využívají řídké reprezentace signálů (viz kapitola 5). Na následujících řádcích je vybráno a stručně popsáno několik stěžejních prací, které se věnují této problematice.

2.2.1 A Constrained Matching Pursuit to Audio Declipping

V tomto článku z roku 2011 se poprvé objevil algoritmus pro declipping využívající řídkou reprezentaci signálu [17]. Algoritmus je zaměřený na obecný zvukový signál, u kterého jsou dopředu známy clippované vzorky. K nalezení řídkých řešení byl na každý blok zpracovávaného signálu použitý omezený ortogonální MP algoritmus (Constrained Orthogonal Matching Pursuit). Bloky jsou zpracovávány metodou OLA (Overlap-add approach), tedy jednotlivé bloky se překrývají, a jako slovník byla použita redundantní diskrétní kosinová transformace (DCT).

2.2.2 Consistent Iterative Hard Thresholding for Signal Declipping

Další práce, která se opět přímo zaměřuje na využití řídkých reprezentací pro vyřešení úlohy declippingu pochází z roku 2013 [18] a je dílem kolektivu autorů kolem Srđana Kitiće, jehož příspěvek je v této oblasti značný. Je zde představen algoritmus *Iterative Hard Thresholding* s rozšířením o omezující podmínky pro účely declippingu nazvaný *Iterative Hard Thresholding – Declipping Constraints* známý také pod zkratkou IHT-DC. Algoritmus podává lepší výsledky než COMP algoritmus, protože pro výpočet využívá i informace ze zkresleného signálu, jako například úroveň prahu oříznutí jednotlivých vzorků.

2.2.3 Declipping of Audio Signals Using Perceptual Compressed Sensing

Tento článek [29] také pochází z roku 2013, ovšem paradoxně zde žádné komprimované snímání neprobíhá. Důležitým poznatkem ale je, že pro restauraci signálu je využito i psychoakustického modelu resp. první vrstvy standardu MPEG-1. Psychoakustický model je blíže rozebrán v kapitole 3.

Řešení úlohy declippingu probíhá pomocí ℓ_1 -minimalizace a jako slovník je použita diskrétní fourierova transformace (DFT). Pro blokové zpracování je na rozdíl od předchozích metod místo obdélníkového okna použito okno Hannovo.

Algoritmus nejprve vypočítá okamžitý globální maskovací práh aktuálního zpracovaného bloku podle standardu MPEG-1 Layer 1. Globální maskovací práh se získá po identifikaci tónových a šumových složek signálu daného bloku a následným výpočtem lokálního maskovacího prahu s ohledem na kmitočtově závislý absolutní práh slyšení. Globální maskovací práh je poté použitý pro následující blok signálu v ℓ_1 minimalizační úloze, kde místo hledaného řídkého vektoru \mathbf{z} vystupuje součin $\mathbf{P}\mathbf{z}$. Matice \mathbf{P} obsahuje na hlavní diagonále reciproké hodnoty okamžitého maskovacího prahu, které tedy označují relativní vjemovou důležitost jednotlivých komponent signálu. Použití této váhové matice \mathbf{P} je z důvodu upřednostnění použití složek, které mají vyšší maskovací práh.

Tento přístup má klady v tom, že dokáže potlačit výrazně slyšitelné složky, které nejsou obsaženy v původním signálu, ale mohly by se objevit s signálu rekonstruovaném. Navíc je podpořena rekonstrukce signálových složek, které jsou z pohledu vnímání zvuku člověkem důležité.

2.2.4 Audio Declipping by Cosparse Hard Thresholding

Narozdíl od *Iterative Hard Thresholding – Declipping Constraints* algoritmu uvedeného v [18], ze kterého tento algoritmus vychází, je zde namísto syntetizujícího modelu použit model analyzující. Tento algoritmus z roku 2014 je nazvaný *Cosparse Declipping by Hard Thresholding*, zkráceně CoDecHT [19].

S analyzujícím modelem ovšem přichází komplikace s výpočtem projekce, která má v tomto případě výpočetní složitost *NP-hard*. Autoři ale tento problém efektivně obešli jiným přístupem, a použili metodu založenou na *Alternating Direction Method of Multipliers*.

Tato metoda na základě výsledků měření ukázala, že je účinnější a oproti algoritmu IHT-DC, ze kterého autoři vycházeli.

2.2.5 Sparsity and Cosparsity for Audio Declipping: A Flexible Non-convex Approach

V roce 2015 byl publikován další článek s novou metodou rekonstrukce audiosignálu založenou na řídkých reprezentacích signálů [20].

V této práci, jejímiž autory jsou Kitić, Bertinová a Gribonval je představen algoritmus založený na nekonvexní heuristice. Předností je možnost použití jak na syntetizujícím, tak analyzujícím modelu signálu.

Algoritmy jsou nazvané obecně *SPADE* (SParse Audio DEclipper) a na základě toho, zda pracují se syntetizujícím či analyzujícím modelem, jsou pak odlišeny jako *S-SPADE* resp. *A-SPADE*. Autoři uvádějí, že v případě S-SPADE algoritmu musí být projekce prováděna iterativně. To způsobuje násobně vyšší výpočetní náročnost v porovnání s algoritmem A-SPADE.

Oba modely jsou shodné, pokud jsou jejich lineární operátory čtvercové a navzájem inverzní matice (tj. když matice \mathbf{A} je báze) pro které platí $\mathbf{A} = \mathbf{D}^{-1}$. Pro matici \mathbf{A} se obvykle používá pojem analyzující operátor a pro matici \mathbf{D} syntetizující operátor, či slovník. Důležitou poznámkou je, že pro oba modely je vhodné, pokud operátor \mathbf{A} tvoří těsný frame.

Jako časově-kmitočtový operátor byla zvolena diskrétní gaborova transformace (DGT), která je taktéž známá pod pojmem krátkodobá fourierova transformace (STFT). Pro váhování bylo použito Hammingovo okno s délkou 1024 vzorků a překryvem 75 %.

2.2.6 Nejnovější příspěvky v oblasti SPADE algoritmů

Aktuálním příspěvkem v této oblasti je práce Pavla Závišky z roku 2018 [21], který zjistil, že se autoři výše zmíněné práce dopustili drobného omylu při řešení optimalizační úlohy, a navrhnul úpravu algoritmu S-SPADE.

Dokázal, že lze snížit výpočetní náročnost algoritmu tak, že nyní je na jednu iteraci stejná pro A-SPADE i S-SPADE. Odstranil tak zjevný nedostatek algoritmu S-SPADE uvedeného v [20]. Použitím zjednodušené *projekce* bylo možné snížit celkový počet iterací algoritmu potřebných ke konvergenci, a tím pádem jej celý urychlit. Na druhou stranu, došlo ke snížení výsledné kvality rekonstruovaného signálu.

O několik měsíců později byly od stejného autora a týmu jeho kolegů uvedeny další články. Ve studii s názvem *S-SPADE Done Right: Detailed Study of the Sparse Audio Algorithms* [22] je upraven algoritmus S-SPADE, který je zde odvozen pomocí tzv. ADMM (*Alternating Direction Method of Multipliers*). Ve studii je dokázáno, že původní S-SPADE není ve všech případech přesný protějšek analyzující verze A-SPADE, ale že to platí jen v případech, kdy jsou operátory unitární, tj. $\mathbf{A} = \mathbf{D}^{-1}$.

V článku *A Proper Version of Synthesis-based Sparse Audio Declipper* [23] je uvedena nová verze algoritmu S-SPADE, která řeší neduh algoritmu uvedeného v [21]. Řešení minimalizačního kroku je v tomto případě dosaženo pomocí tzv. *hard-thresholdingu* v Hilbertově prostoru. Výpočetní náročnost je určena především transformací signálu. V tomto případě syntéza a analýza. Nyní algoritmus dosahuje srovnatelné kvality restaurace zvukového signálu jako A-SPADE a dokonce konverguje rychleji.

3 Psychoakustika

Psychologická akustika, zkráceně psychoakustika, se zabývá zkoumáním procesů lidského vědomí v souvislosti s vnímáním resp. působením zvuku. Proces odrazu objektivních fyzikálních vlastností těchto signálů ve vědomí člověka je velmi složitý, na jeho začátku se nalézá percepce jako smyslová podoba či vlastní vjem hudebního signálu [41].

Pro potřeby úlohy declippingu vystupuje jako stěžejní oblast psychoakustiky maskování. Využití těchto poznatků lze využít pro rekonstrukci jen těch znehodnocených vzorků, které jsou percepčně významné a naopak ignorování irelevantních elementů zvukové informace. Tyto a další poznatky jsou shrnuty v psychoakustickém modelu, který je použitý u ztrátové komprese MPEG (zkratka z anglického *Moving Pictures Expert Group*).

Již v článku [45] bylo zjištěno, že vliv implementace psychoakustického modelu do algoritmu pro declipping má velmi příznivý vliv na výsledek declippingu a to jak v termínech subjektivní, tak objektivní kvality restaurovaného zvukového signálu.

V první části této kapitoly je proto vysvětlen pojem maskování a práh slyšení. V další části je podrobněji rozebráno maskování zvuků v kmitočtové i časové oblasti.

3.1 Maskování

Znění určitého tónu nebo hluku vyvolává současně též posuv sluchového prahu směrem k vyšším hodnotám. Vjem dalšího tónu či hluku o nižší intenzitě může být proto zeslaben nebo přímo potlačen. Tento jev se nazývá *maskování*. První zvuk se nazývá *maskující* a druhý, který je potlačen, *maskovaný* [41]. Pro maskující zvuk se také používá termín *masker* a pro maskovaný *maskee*. Rozlišuje se kmitočtové maskování (anglicky *simultaneous masking*) současně znějících zvuků a dočasné maskování (z anglického *temporal masking*). Tyto jevy jsou ale od sebe prakticky neoddělitelné.

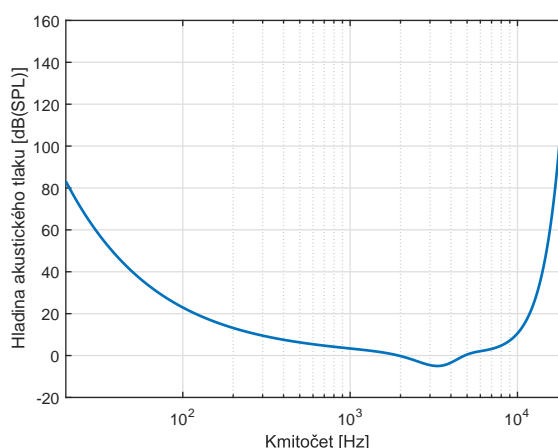
3.1.1 Práh slyšení

Maskování je spjato s kmitočtově závislým sluchovým prahem uvedeným na obrázku 3.1, protože se měří posunem sluchového prahu maskovaného zvuku. Pod úrovní sluchového prahu člověk není schopen vnímat zvukovou informaci. Tyto prahy byly změřeny pro harmonické signály, širokopásmový i úzkopásmový šum a komplexní zvukové signály. Křivka prahu slyšitelnosti T_q byla aproximována pomocí funkce

$$T_q(f) = 3,64 \left(\frac{f}{1000} \right)^{-0,8} - 6,5e^{-0,6 \left(\frac{f}{1000} - 3,3 \right)^2} + 10^{-3} \left(\frac{f}{1000} \right)^4, \quad (3.1)$$

kde f je kmitočet.

Práh slyšitelnosti je definován jako minimální průměrná hladina akustického tlaku, kterou je schopný člověk se zdravým sluchem slyšet. Vztažná hodnota hladiny akustického tlaku $p_0 = 2 \cdot 10^{-5}$ Pa byla zvolena podle hodnoty prahu slyšitelnosti na kmitočtu 1 kHz. Tento práh je závislý na kmitočtu, na charakteru zvuku (čistý tón, širokopásmový šum, řeč, atd.), poslechu ve volném nebo difúzním poli a věku posluchače. S narůstajícím věkem se snižuje schopnost slyšet vysoké kmitočty. Rozdíl je i mezi monaurálním a binaurálním poslechem [42]. Z grafu 3.1 je patrné,



Obr. 3.1: Závislost prahu slyšení na kmitočtu.

že na velmi nízkých a vysokých kmitočtech dochází v nárůstu hodnot prahu slyšení, a naopak kolem kmitočtu 3 kHz, kde je citlivost nejvyšší, se práh dostává do záporných hodnot. To je způsobeno zvolenou vztažnou hodnotou p_0 , která je na kmitočtu 1000 Hz vyšší.

3.1.2 Maskování v kmitočtové oblasti

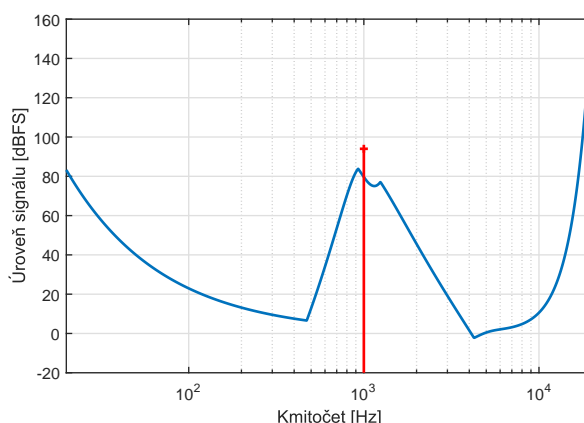
Maskující tón rozkmitá určitou část bazilární membrány, která ale přesně neodpovídá mozku vyhodnocenému kmitočtu tónu odpovídajícímu konkrétnímu bodu, nýbrž k podráždění smyslových buněk dochází na delším úseku bazilární membrány.

Smyslové buňky ve vnitřním uchu se při podráždění ocitnou po dobu několika milisekund ve stavu, kdy nejsou schopny přijmout další podráždění. Proto zvuk, který zasytí vzruchy příslušnou oblast bazilární membrány, znemožní slyšení jiného a slabšího zvuku, který ve stejný okamžik podráždí stejnou zónu. Tomuto jevu se říká spektrální neboli kmitočtové maskování [42]. Největší maskování překvapivě nastává při souhlasu resp. těsné blízkosti kmitočtů, protože identifikaci maskovaného tónu zde usnadňují vznikající zázněje s maskujícím tónem [41].

Maskovací zvuk rozšiřuje oblast neschopnosti vnímat zvuk o oblast, která je popsána tzv. *maskou*. Oblast neschopnosti vnímat zvuk je pak dána příspěvkem

všech oblastí pod maskami aktuálních maskovacích zvuků a oblasti pod prahem slyšení. Tvar masky je téměř neměnný v závislosti na kmitočtu maskovacího zvuku, pokud je maska vynášena na kmitočtové ose barkových pásmech. Směrem k nižším kmitočtům je strmost průběhu větší, než směrem k vyšším kmitočtům. Tvar masky se také odvíjí od intenzity maskovacího zvuku, kdy s vyšší intenzitou se maska rozšiřuje zejména směrem k vysokým kmitočtům.

Obrázek 3.2 zobrazuje aproximovanou křivku masky, kterou vytvoří harmonický signál o kmitočtu 1 kHz a hladině 94 dB(SPL). Křivka masky je zde sečtená s křivkou prahu slyšitelnosti a tvoří tak *maskovací práh*.



Obr. 3.2: Maskovací práh harmonického signálu o kmitočtu 1 kHz a hladině 94 dB(SPL).

Je-li čistý tón maskován šumem s konstantní energií v celém kmitočtovém pásmu (například bílý šum), podílí se na vlastním maskování pouze určité pásmo tohoto šumu. Šířka tohoto pásma se nazývá *kritická šířka pásma* a mění se v závislosti na kmitočtu. Kritická šířka pásma je těsně spjatá s fyziologickými vlastnostmi sluchu. Jednotlivým kritickým pásmům označovaným jako *barky* byly přiřazeny na způsob sady pásmových filtrů konkrétní střední kmitočty a šířky jednotlivých pásem. Nevýhodou se ukázala nízká přesnost na nízkých kmitočtech, proto byla později stanovena ekvivalentní obdélníková pásma *ERB* (*Equivalent Rectangular Bandwidth*) navržená Moorem a Glasbergem [41].

3.1.3 Maskování v časové oblasti

Maskování se nevyskytuje pouze při současném znění dvou zvuků, ale také jistou dobu po odeznění maskovacího zvuku, kde se hovoří o *post-masking*, a dokonce i krátký okamžik před zazněním maskovacího zvuku, a tehdy se nazývá *pre-masking*.

V druhém zmíněném případě není myšlen okamžik vstupu zvukového signálu do sluchového aparátu, ale okamžik percepce zvuku v mozku [43].

Příčinou post-masking je neschopnost smyslových buněk na bazilární membráně přijmout okamžitě další informace a jeho doba trvání se pohybuje od 100 do 200 ms. Proto je dominantní jako maskovací efekt v časové rovině a pre-masking je v některých případech dokonce ignorován.

Pre-masking je způsoben nástupem maskovacího zvuku a souvisí s rychlostí zpracování silnějšího a slabšího podnětu. Maskovací zvuk se nedokáže naplno projevit okamžitě, ale potřebuje jistý čas pro vytvoření jeho vjemu v mozku. Maskovací zvuky s vysokou intenzitou mají proto kratší nástup (tzv. *build-up time*), než maskovací zvuky s nízkou intenzitou. Pre-masking trvá přibližně 20 ms a je zde pouze malá závislost na hladině maskovacího zvuku [43].

4 Hodnocení kvality restaurace signálů

K vyhodnocení kvality algoritmů pro declipping se používají různé druhy testů. Jak pro srovnání kvality signálu před clippingem a po rekonstrukci, tak pro vzájemné porovnání kvality různých algoritmů. Existují testy objektivní, které poskytují přesné a vyčíslitelné výsledky, stejně tak jako subjektivní poslechové testy pro skupinu posluchačů. V této kapitole budou popsány dvě nejrelevantnější objektivní metody vyhodnocení kvality restaurace clipovaných signálů. A to *SNR* (resp. *SDR*) a *PEMO-Q*. Ze subjektivních testů bude uvedena metoda zvaná *MUSHRA*.

4.1 Objektivní metody

Objektivní metody testování kvality zvukových signálů poskytují přesné a nezájaté výsledky v porovnání s testy subjektivními, kde hraje roli vliv prostředí a aktuální psychické i fyzické rozpoložení účastníků testu. Předností objektivních testů je jejich poměrně snadná implementace i rychlost vyhodnocení výsledků. Skutečnost, že objektivní testy neudávají relevantní informace o subjektivní vjemové kvalitě se snaží některé metody kompenzovat pomocí různých technik tak, aby se minimalizoval rozdíl mezi objektivním výsledkem a subjektivním vjemem kvality zvukového signálu. V následujících podkapitolách jsou uvedeny dva různé testy, které jsou vhodné pro měření kvality restaurace poškozených signálů.

4.1.1 SDR

SDR (*Signal to distortion ratio*) neboli poměr úrovně signálu k úrovni zkreslení je nejfrekventovaněji používaný test pro zjištění kvality restaurovaných zvukových signálů. Obecně poskytuje bezrozměrnou informaci o poměru výkonu signálu k výkonu zkreslení v daném signálu [44]. Definice SDR je shodná s definicí SNR (*Signal to noise ratio*) neboli poměrem úrovně signálu k úrovni šumu.

V rámci hodnocení restaurace ovšem hodnota SDR představuje odstup restaurovaného signálu od originálního signálu. Výsledkem je hodnota v decibelech a platí, že čím je tato hodnota vyšší, tím si jsou signály podobnější.

Užitečné je používat rozdíl SDR značený ΔSDR , který je vypočten jako

$$\Delta\text{SDR} = \text{SDR}(\mathbf{s}, \mathbf{s}') - \text{SDR}(\mathbf{s}, \mathbf{s}_c), \quad (4.1)$$

kde je $\text{SDR}(\mathbf{u}, \mathbf{v})$ vypočteno jako

$$\text{SDR}(\mathbf{u}, \mathbf{v}) = 10 \log_{10} \left(\frac{\|\mathbf{u}\|_2^2}{\|\mathbf{u} - \mathbf{v}\|_2^2} \right) \quad (4.2)$$

s tím, že s je původní signál, s_c je clipovaný signál a s' je declipovaný signál [29]. Nedostatek této metody spočívá v tom, že lze získat pouze informaci o podobnosti dvou signálů, ale bez jakékoliv záruky, že signál bude i subjektivně lepší na poslech.

4.1.2 PEMO-Q

Metoda PEMO-Q (*Perception Model – Quality Assessment*) je objektivní metoda měření kvality zvukového signálu. Tato metoda je založena na hodnocení odchylky vnitřní reprezentace mezi referenčním a testovaným zvukem [47].

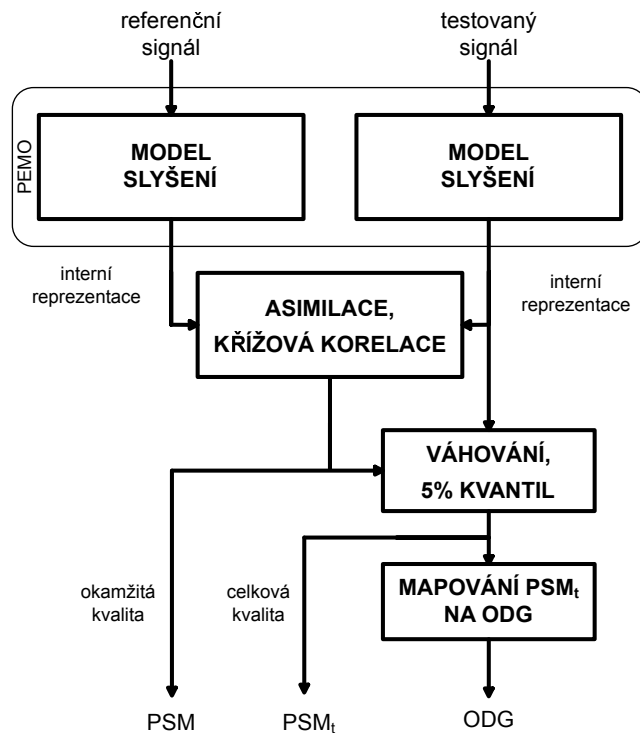
Cílem použitého sluchového model PEMO je simulace transformace akustického podnětu lidským uchem na nervové vzruchy, které jsou v mozku následně zpracovávány. Z výsledku křížové korelace mezi oběma signály je vypočítána hodnota PSM (*Perceptual Similarity Measure*), která představuje okamžitou kvalitu v čase. PSM_t jakožto druhá výstupní hodnota je 5% kvantil z váženého průměru křížové korelace a udává celkovou vnímanou kvalitu signálu.

Třetím výstupem jsou hodnoty na stupnici ODG (*Objective Difference Grade*) převedené z PSM_t. Nacházejí se v rozsahu 0 až −4. Stupnice ODG, která je zakotvena v ITU-R [45], indikuje zhoršení objektivní kvality signálu a její škála má k jednotlivým hodnotám přiřazený slovní popis, jak je uvedeno v následující tabulce.

Tab. 4.1: Pětibodová škála hodnocení poškození zvukových signálů ODG

ODG	poškození
0	neznatelné (<i>imperceptible</i>)
-1	znatelné, ale nikoliv nepříjemné (<i>perceptible but not annoying</i>)
-2	trochu nepříjemné (<i>slightly annoying</i>)
-3	nepříjemné (<i>annoying</i>)
-4	velmi nepříjemné (<i>very annoying</i>)

Pro porozumění činnosti metody PEMO-Q je na obrázku 4.1 uvedeno blokové schéma.



Obr. 4.1: Blokové schéma metody PEMO-Q.

4.2 Subjektivní metody

Subjektivní metody měření kvality lze rozdělit do dvou kategorií. První z nich jsou metody založené na relativní preferenci a druhá kategorie využívá absolutní hodnocení kvality. K provedení testů je nutný jistý minimální počet respondentů, stabilita prostředí a vhodně navržený test, aby byla zajištěna dostatečná relevance výsledku. Výhodou tohoto typu testování je fakt, že objektivní metody nemusí vůbec korelovat se subjektivním dojmem posluchače, který je zejména při declippingu rozhodující. Nevýhodou je časová náročnost testování, ať už se jedná o přípravu, samotný test, či následné vyhodnocení nasbíraných dat.

4.2.1 MUSHRA

Metoda MUSHRA (*MULTI Stimulus test with Hidden Reference and Anchor*) čili více-stimulový test se skrytou referencí a kotvou je standardizovanou metodou pro subjektivní testování. Je vhodná pro případy, kdy se očekává jistá snížená kvalita a i pozornější posluchač ji může odhalit. Při testování je posluchači předloženo několik nahrávek včetně známé reference (čistý signál) a skryté reference. Mezi jednotlivými nahrávkami je skryta i tzv. *kotva*, což je nahrávka s nejvíce zhoršenou kvalitou. Ta slouží pro lepší vymezení hodnotící škály mezi kotvou a referencí [46].

Hodnoticí škála je plynulá stupnice s 5 definovanými body. V praxi se používá celých čísel od 0 do 100. Získané hodnoty se vztahují pouze na testovaný vzorek posluchačů. Ve statistice se tento vzorek posluchačů označuje jako *výběr*.

Tab. 4.2: Stupnice hodnocení metody MUSHRA

hodnocení	kvalita
100–80	vynikající (<i>excellent</i>)
80–60	dobrá (<i>good</i>)
60–40	dostačující (<i>fair</i>)
40–20	špatná (<i>poor</i>)
20–0	nepříjemná (<i>bad</i>)

5 Řídké reprezentace signálů

Řídké reprezentace signálů lze vnímat jako jeden ze způsobů řešení nedourčeného systému lineárních rovnic. Pro takové rovnice obecně existuje nekonečně mnoho řešení, mezi kterými jsou i řešení řídká, tzn. řešení, která obsahují velmi málo nenulových proměnných.

Tato řešení nachází svá uplatnění například v oblasti zpracování obrazu (komprimované snímání [24]), a také při restauraci znehodnocených zvukových signálů. V této podoblasti se může jednat o doplňování chybějících informací v signálu, tzv. inpainting [25], declipping či odšumování [26, 18, 20].

V úvodu této kapitoly je věnován prostor popisu a vysvětlení základních pojmů a nastínění problematiky hledání řídkého řešení systémů lineárních rovnic. V části 5.4 se nachází formulace základní úlohy a následně upřesňující podmínky, tak aby bylo nalezeno jedinečné řídké řešení problému. V další části je vysvětleno použití ℓ_1 -minimalizace, respektive ℓ_1 -relaxace.

5.1 Výchozí základy a značení

Nejprve je nutné zadefinovat a vysvětlit stěžejní pojmy, které budou v následujících kapitolách použity.

Skalární veličiny jako např. s , S jsou označeny kurzívou, vektory tučným řezem, jako např. \mathbf{y} . Matice jsou taktéž tučným řezem, pouze s tím rozdílem, že jsou označeny verzálkou. Vektory jsou uvažovány jako sloupcové a pouze s konečným počtem prvků. Indexace prvků vektorů začíná jedničkou, tzn. $\mathbf{x} = [1, 2, 3, \dots]^\top$. Počet prvků množiny neboli kardinalita množiny bude značen stejně jako absolutní hodnota; tj. např. $|\{-7, 2, 4, 5\}| = 4$. Jádro lineárního zobrazení určeného maticí \mathbf{A} se bude značit $\ker \mathbf{A}$.

Pojem nosič vektoru \mathbf{x} znamená množinu indexů, na kterých má daný vektor nenulové hodnoty. Pro nosič vektoru se používá označení $\text{supp}(\mathbf{x})$ (z anglického *support*) a obecná definice zní $\text{supp}(\mathbf{x}) = \{i \mid x_i \neq 0\}$ [30].

Vektorové prostory jsou označeny jako \mathbb{V} , \mathbb{C}^N atp. a jsou uvažovány jako neprázdné. Systémy generátorů vektorových prostorů včetně bází mají shodné označení jako matice, tj. tučné verzálky \mathbf{P} , \mathbf{R} atd. \mathbf{I} značí jednotkovou matici [27].

5.2 Normy vektorů

Norma vektoru \mathbf{x} značená $\|\mathbf{x}\|$ je reálná funkce, která každému nenulovému vektoru přiřazuje kladné reálné číslo, kterým je popsána jeho tzv. velikost. Nejčastěji

používanou normou je euklidovská norma. Takzvaná ℓ_p -norma vektoru $\mathbf{x} \in \mathbb{C}^N$ je definována jako:

$$\begin{aligned}\|\mathbf{x}\|_p &:= \left(\sum_{i=1}^N |x_i|^p \right)^{\frac{1}{p}} \quad \text{pro } 1 \leq p < \infty, \\ \|\mathbf{x}\|_p &:= \sum_{i=1}^N |x_i|^p \quad \text{pro } 0 < p < 1, \\ \|\mathbf{x}\|_\infty &:= \max |x_i|, \\ \|\mathbf{x}\|_0 &:= |\text{supp}(\mathbf{x})|.\end{aligned}\tag{5.1}$$

O normu se přísně vzato jedná pouze v případě $1 \leq p \leq \infty$. Pro zjednodušení však bude pro všechna p použito jednotně označení ℓ_p -norma. ℓ_1 -norma představuje součet absolutních hodnot prvků vektoru, ℓ_0 -norma počet nenulových složek vektoru a konečně ℓ_∞ -norma vrací prvek vektoru s největší hodnotou [27].

5.3 Řídkost vektoru

O vektoru se dá říci, že je řídký, pokud má převážná část jeho prvků nulovou hodnotu [28]. Vektor $\mathbf{x} \in \mathbb{C}^N$ je k -řídký (k -sparse), pokud platí

$$\|\mathbf{x}\|_0 \leq k.\tag{5.2}$$

Jinými slovy, k -řídký vektor je tedy takový, který má nejvýše k nenulových složek. Relativní řídkost vektoru \mathbf{x} je poměr $\frac{k}{N}$, kde N je celková délka vektoru \mathbf{x} .

5.4 Řídká řešení systémů lineárních rovnic

Při hledání řídkého řešení soustavy lineárních rovnic $\mathbf{Ax} = \mathbf{y}$, se dá na tuto problematiku pohlížet jako na hledání neznámého vektoru \mathbf{x} , který by měl být co nejřidší, tzn. měl by obsahovat co největší počet složek s nulovou hodnotou. Tuto úlohu je možno formulovat jako

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{vzhledem k } \mathbf{Ax} = \mathbf{y},\tag{5.3}$$

kde je předem známý vektor $\mathbf{y} \in \mathbb{C}^m$ a matice $\mathbf{A} \in \mathbb{C}^{m \times N}$. Vektor \mathbf{y} může být vstupní signál, výsledek pozorování či měření. Do úvahy jsou vzaty pouze případy, kdy platí $m < N$, resp. $m \ll N$ a matice \mathbf{A} je plně řádkové hodnosti. Matice \mathbf{A} se nejčastěji nazývá *slovník*, přičemž jednotlivé sloupce matice se nazývají *atomy*. Lze se setkat i s označením matice jako *reprezentační systém*.

Všechna \mathbf{x} , která splňují $\mathbf{Ax} = \mathbf{y}$, se nazývají přípustná řešení, neboli přípustné reprezentace vektoru \mathbf{y} . Z lineární algebry je známo, že za výše uvedených podmínek

na matici \mathbf{A} je přípustných řešení nekonečně mnoho a tvoří afinní prostor. V následujících odstavcích je věnována pozornost požadavkům kladených na matici \mathbf{A} , které zaručí možnost „rekonstrukce“ (*sparse recovery*) původního \mathbf{x} z \mathbf{y} .

V četných případech, kdy je signál zašuměn je třeba zavést malou odchylku při výpočtu $\mathbf{Ax} = \mathbf{y}$, protože šum v signálu může způsobit odchýlení od přesného měření. Nyní tedy úloha vypadá takto:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{vzhledem k } \|\mathbf{Ax} - \mathbf{y}\|_p \leq \delta, \quad (5.4)$$

kde se nejčastěji uvažuje $p = 2$ [30].

5.4.1 Postačující podmínky jednoznačného řešení

Nyní budou popsány postačující podmínky, za kterých je možné dokonale restaurovat k -řádký signál. Někdy nalezení řídkého řešení není jednoznačné, či dokonce řešení vůbec neexistuje. Na druhou stranu postačující podmínky jsou schopny určit, zda řídké řešení existuje, a případně jestli je také jednoznačné a nejřidší možné. Slovník zde nabývá vysoké důležitosti, protože od jeho vlastností se odvíjí podmínky pro dokonalou rekonstrukci původního signálu [29].

Číslo *spark* je definováno jako nejmenší počet sloupců matice \mathbf{A} , které jsou lineárně závislé. Lze jej zapsat jako

$$\text{spark}(\mathbf{A}) = \min_{\mathbf{z} \in \ker(\mathbf{A}), \mathbf{z} \neq 0} \|\mathbf{z}\|_0. \quad (5.5)$$

Čím je číslo *spark* menší, tím řidší musí vektor \mathbf{x} být, aby mohla být splněna jedinečnost tohoto řešení. Pro řešení soustavy $\mathbf{Ax} = \mathbf{y}$ pak platí, že pokud řešení \mathbf{x} splňuje podmínku

$$\|\mathbf{x}\|_0 < \frac{\text{spark}(\mathbf{A})}{2}, \quad (5.6)$$

pak je toto řešení jedinečné a zároveň nejřidší možné. Bohužel výpočet čísla $\text{spark}(\mathbf{A})$ je srovnatelně náročný jako řešení problému (5.3), a proto se v praxi v podstatě nepoužívá. Navíc, podmínka (5.6) poskytuje pouze postačující podmínku, takže reálně může existovat řešení, které ačkoliv je nejřidší, nemusí tuto podmínku splňovat.

Vzájemná koherence matice \mathbf{A} je definována jako největší absolutní normalizovaný skalární součin dvou různých sloupců matice \mathbf{A} ,

$$\mu(\mathbf{A}) = \max_{1 \leq j, k \leq N, j \neq k} \frac{|\mathbf{a}_j^\top \mathbf{a}_k|}{\|\mathbf{a}_j\|_2 \cdot \|\mathbf{a}_k\|_2}, \quad (5.7)$$

kde \mathbf{a}_j označuje j -tý atom (sloupec) matice \mathbf{A} . Tímto způsobem lze zjistit „míru lineární závislosti“ mezi sloupci matice. Mezi vzájemnou koherencí a číslem *spark* platí pro libovolnou matici \mathbf{A}

$$\text{spark}(\mathbf{A}) \geq 1 + \frac{1}{\mu(\mathbf{A})}. \quad (5.8)$$

Ze vztahu (5.8) vyplývá, že pokud se pro soustavu $\mathbf{Ax} = \mathbf{y}$ nalezne řešení \mathbf{x} splňující

$$\|\mathbf{x}\|_0 < \frac{1}{2} \left(1 + \frac{1}{\mu(\mathbf{A})} \right), \quad (5.9)$$

pak je \mathbf{x} nutně nejřidší možné a také jedinečné. Navíc tohoto řešení lze dosáhnout i ℓ_1 -minimalizací. Z výše uvedeného vztahu lze vyčíst, že pokud se bude vzájemná koherence $\mu(\mathbf{A})$ blížit nule, pravá strana rovnice poroste nade všechny meze, a proto je tendence používat maximálně nekoherentní slovníky [30].

5.5 ℓ_1 -relaxace

Vzhledem k tomu, že v praxi je výpočet přesného řešení problému (5.3) výpočetně a časově velmi náročné, je třeba udělat kompromis a obětovat jistou míru přesnosti, a získat tak výsledek v přijatelném čase [30].

K tomu je třeba vzít v úvahu, že ℓ_0 -norma není konvexní a proto ani nelze použít metody a algoritmy konvexní optimalizace. Existuje zde ale eventualita v podobě nahrazení ℓ_0 -normy první nejbližší ℓ_p -normou, která už konvexní je. Jelikož platí, že ℓ_p -normy jsou konvexní pro $p \geq 1$, nabízí se norma ℓ_1 pro získání alespoň přibližného řešení problému (5.3) a tím pádem řešit úlohu

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{vzhledem k } \mathbf{Ax} = \mathbf{y}. \quad (5.10)$$

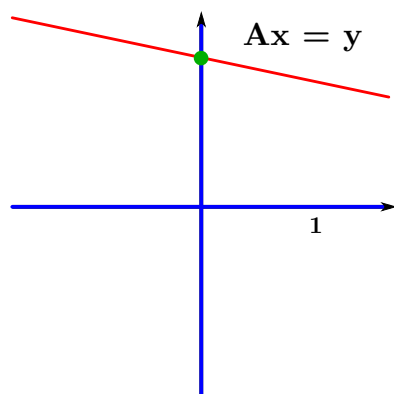
Stejně jako v případě ℓ_0 -normy i zde je více odpovídající úloha, která počítá se zašuměným signálem a povoluje tak jistou odchylku (neboli relaxaci)

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{vzhledem k } \|\mathbf{Ax} - \mathbf{y}\|_2 \leq \delta. \quad (5.11)$$

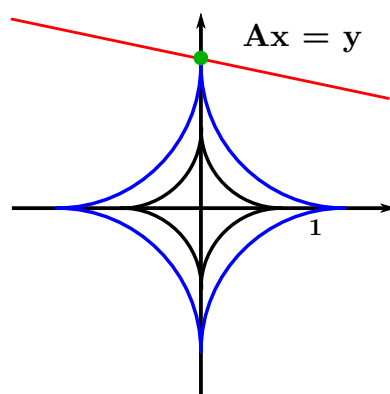
Tento problém je znám pod zkratkou LASSO (*Least Absolute Shrinkage and Selection Operator*). Výpočet tohoto problému sice nezaručuje nejřidší možné řešení, na druhou stranu v praxi se ukázalo, že ve většině případů se řešení ℓ_1 -minimalizace shoduje s řešením ℓ_0 -minimalizace. V jistých případech může ℓ_1 -minimalizace selhat, protože nalezne nekonečně mnoho řešení s minimální normou, ale pouze dvě „krajní“ jsou řídka.

V zobrazení 5.1 je ilustrováno řešení úlohy $\min_{\mathbf{x}} \|\mathbf{x}\|_p$ vzhledem k $\mathbf{Ax} = \mathbf{y}$ v \mathbb{R}^2 pro $p = 0; 0,5; 1$ a 2 jako názorný příklad ekvivalence. Prostor všech přípustných řešení reprezentuje červená přímka, která zde představuje nadrovinu soustavy $\mathbf{Ax} = \mathbf{y}$. Z obrázků 5.1(a) a 5.1(c) je vidět shoda řešení problému (5.3) a (5.10). Při použití euklidovské normy nelze dojít k jedinečnému řešení, které by bylo řídke.

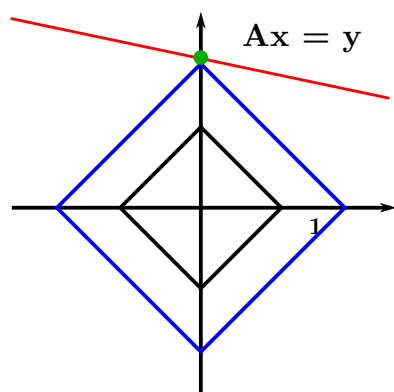
Nalezení podmínek ekvivalence ℓ_0 a ℓ_1 minimalizace v současné době stále není uzavřenou kapitolou. Mezi již existující podmínky patří například vlastnost nulového prostoru *NSP* (*Null Space Property*) nebo vlastnost zeslabené isometrie *RIP* (*Restricted Isometry Property*) [27].



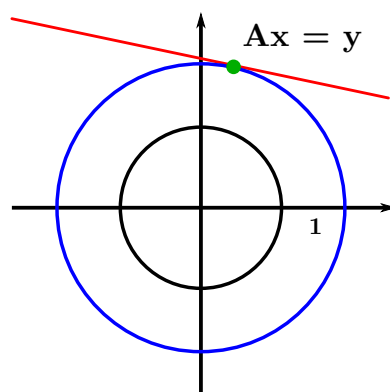
(a) Norma ℓ_0 .



(b) Norma $\ell_{0,5}$.



(c) Norma ℓ_1 .



(d) Norma ℓ_2 .

Obr. 5.1: Zobrazení jednotkové koule pro různé normy.

6 Báze a frame

Tato kapitola popisuje pojmy týkající se vektorových prostorů jako je generátor vektorového prostoru, báze a frame. Tyto pojmy jsou zavedeny proto, že představují reprezentační systém sloužící k práci s nedourčenými systémy lineárních rovnic. V souvislosti s tím je zde zdůvodněno i zavedení gaborovského systému resp. transformace, která v této práci nachází využití pro řešení úlohy declippingu jako redundantní reprezentační systém.

6.1 Generátory vektorového prostoru

Vektorový prostor (VP), který je zde uvažován s konečnou dimenzí, je algebraická struktura splňující známé axiomy, jehož prvky jsou vektory [31].

Systém generátorů vektorového prostoru \mathbb{V} je podmnožina vektorů \mathbf{E} ve \mathbb{V} , která jej generuje, čímž se rozumí, že každý vektor $\mathbf{x} \in \mathbb{V}$ je jistou lineární kombinací generátorů \mathbf{E} . Generátorů \mathbb{V} může být i více než je dimenze n , a pak vektor $\mathbf{x} \in \mathbb{V}$ lze napsat jako vzájemně různé lineární kombinace, a tak jeden vektor může mít více reprezentací. Taková situace se nazývá *nedourčenost*. Pokud \mathbf{x} lze v systému jeho generátorů $\mathbf{E} = \{e_1, e_2, \dots, e_n\}$ zapsat jako

$$\mathbf{x} = c_1 \mathbf{e}_1 + c_2 \mathbf{e}_2 + \dots + c_n \mathbf{e}_n = \mathbf{E} \mathbf{c}, \quad (6.1)$$

pak skaláry c_i představují souřadnice \mathbf{x} v \mathbf{E} a při bližším zkoumání tento zápis koreponduje s tzv. aditivním způsobem modelování signálu, kdy signál \mathbf{x} je považován za superpozici (lineární kombinaci) dílčích signálů

$$\mathbf{x} = \sum_{i=1}^m c_i \mathbf{e}_i. \quad (6.2)$$

Nutno podotknout, že i -tý koeficient odpovídá i -tému sloupci matice \mathbf{E} [32].

6.2 Báze vektorového prostoru

Bází VP se rozumí minimální systém jeho generátorů. Je to množina lineárně nezávislých vektorů, jejichž lineární kombinací lze dosáhnout na libovolný vektor v uvažovaném VP. V konečně rozměrném prostoru dimenze n je bází každá množina obsahující n lineárně nezávislých vektorů. Je-li $\mathbf{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_n\}$ bází, pak každý prvek $\mathbf{x} \in \mathbb{V}$ lze vyjádřit pomocí jednoznačných souřadnic c_i v oné bází jako

$$\mathbf{x} = \sum_{i=1}^n c_i \mathbf{b}_i = \mathbf{B} \mathbf{c}. \quad (6.3)$$

Má-li jeden VP více bází, potom všechny mají stejný počet generujících vektorů. Nejčastěji se používají báze ortogonální a ortonormální. Ortogonální báze je taková, kdy pro libovolné dva vektory z báze $\mathbf{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_n\}$ platí

$$\langle \mathbf{b}_i, \mathbf{b}_j \rangle = 0, \langle \mathbf{b}_i, \mathbf{b}_i \rangle \neq 0, \quad (6.4)$$

tedy všechny dvojice bázových vektorů jsou na sebe kolmé. V ortonormální bázi navíc oproti (6.4) platí pro všechny její prvky $\|\mathbf{b}_i\| = 1$, neboli $\mathbf{B}^* = \mathbf{B}^{-1}$ [32].

Z pohledu numerické stability je ideální používat ortogonální báze. Při zvyšování počtu generátorů se zhoršuje podmíněnost matice \mathbf{A} , tzn. velké změny v řešení \mathbf{x} mohou vést k pouze malým změnám \mathbf{y} . Na druhou stranu se zvyšuje řídkost řešení uvažovaného systému $\mathbf{A}\mathbf{x} = \mathbf{y}$ [33].

6.3 Frame

Když je počet generátorů větší, než je dimenze prostoru, můžeme stále reprezentovat jakýkoliv vektor ve VP, generátory ale už budou nutně lineárně závislé. Taková množina vektorů se nazývá *frame*, nebo přesněji *konečný frame* ve \mathbb{V} . Toto redundantní vyjádření může být pro reprezentaci signálu i výhodou, protože framy jsou oproti bázím flexibilnější. Nevýhodou jsou náročnější výpočetní metody a riziko numerické nestability [27].

Nejvýše spočetná množina vektorů $\{\varphi_k\}_{k \in \mathbb{I}}$ v prostoru \mathbb{V} v něm tvoří frame, pokud existují konstanty $0 < A \leq B < \infty$ takové, že platí

$$A\|\mathbf{x}\|^2 \leq \sum_{k \in \mathbb{I}} |\langle \mathbf{x}, \varphi_k \rangle|^2 \leq B\|\mathbf{x}\|^2, \forall \mathbf{x} \in \mathbb{V}. \quad (6.5)$$

Prvky framu φ_k se obvykle nazývají *atomy*. Konstanty A a B se pak nazývají meze framu. Optimální dolní mez se definuje jako supremum ze všech dolních mezí a optimální horní jako infimum z horních mezí.

Je dokázáno, že každá spočetná množina vektorů je frame pro jeho lineární obal. Také platí, že v konečné dimenzi z každého framu lze vybrat bázi [34], [35]. Jednou ze stěžejních oblastí využití framů je právě řídká reprezentace signálů, kdy je snahou přizpůsobit atomy framu co nejvíce signálu, a tím získat úspornou reprezentaci.

Framy, pro které platí $A = B$, se nazývají *těsné* (z anglického *tight frame*). Stejná rovnost platí i pro ortogonální báze se shodně velkými prvky, takže z tohoto pohledu jsou rovnocenné, což mimo jiné přináší jednu obrovskou výhodu oproti framům, které nejsou těsné.

Zvláštním případem je pak tzv. 1-těsný frame, který se často nazývá *Parsevalův těsný frame* (Parseval tight frame – PTF), kdy platí dokonce $A = B = 1$. Další skupinou jsou framy, pro které platí, že všechny jejich prvky mají stejnou normu

(Equal-norm frames – ENF) a unitární framy (Unit-norm frames – UNF), jejichž všechny prvky mají normu rovnou 1.

6.4 Gaborův Frame

Zavedení Gaborova systému souvisí s faktem, že Fourierova transformace pracuje s harmonickými funkcemi, které mají globální dosah. To znamená, že jeden významný koeficient ovlivní podobu celého signálu. To však nekoresponduje s tím, jak funguje lidské vnímání zvuků – zjednodušeně, člověk vnímá jejich kmitočtovou strukturu, ale tu dokáže zároveň umístit v čase [27].

Časově-kmitočtová lokalizace získaná vyříznutím signálu a provedením FT není optimální, protože vyříznutí odpovídá násobení signálu obdélníkovým signálem, jehož podoba v kmitočtové oblasti odpovídá funkci sinc, jejíž hodnota směrem k vysokým kmitočtům klesá velmi pomalu. Limit časově-kmitočtového rozlišení popisuje *Heisenbergův princip neurčitosti*, který říká, že žádný signál nemůže být zároveň koncentrován v čase i v kmitočtu.

Matematická teorie Gaborovy analýzy v $L^2(\mathbb{R})$ je postavena na operátorech translace a modulace. Gaborova analýza se proto zaměřuje na reprezentaci funkce $f \in L^2(\mathbb{R})$ jako superpozici posunutých a modulovaných verzí fixní funkce $g \in L^2(\mathbb{R})$. Soubor takovýchto funkcí

$$\{e^{j2\pi mbx}g(x - na)\}_{m,n \in \mathbb{Z}} \quad (6.6)$$

se nazývá Gaborův systém. Funkce g se nazývá okénková funkce nebo generátor [35]. Cílem snažení je, aby tento systém s funkcí $g \in L^2(\mathbb{R})$ a parametry translace a modulace a a b tvořil frame v prostoru $L^2(\mathbb{R})$.

Tato problematika není dosud zcela vyřešena, ale existuje alespoň nutná podmínka pro existenci Gaborova framu. Ta říká, že pokud $a, b > 0$ a $g \in L^2(\mathbb{R})$, pak gaborovský systém (6.6) může být frame pro $L^2(\mathbb{R})$, pokud platí $ab \leq 1$. Pokud je gaborovský systém frame, pak je možné každý signál rekonstruovat z jeho časově-kmitočtových koeficientů [27].

7 Proximální algoritmy

Pro řešení optimalizačních úloh existuje v současnosti řada algoritmů. V této práci bude nadále věnována pozornost především tzv. *proximálním algoritmům*, které lze využít pro declipping. Dosud žádná metoda nedokáže pokaždé najít v polynomiálním čase přesné řešení, proto je nutné spokojit se s přibližným, tj. aproximativním řešením problémů.

V této části bude vymezen pojem proximální algoritmus a uvedeny proximální operátory pro ℓ_1 -normu a indikátorovou funkci. Dále jsou popsány metody proximálního dělení jako předvoj tří algoritmů, pomocí kterých lze řešit úlohu declippingu. Jedná se o Condatův algoritmus, Douglas-Rachfordův algoritmus a algoritmus, který společně navrhli Chambolle a Pock.

7.1 Vymezení proximálních algoritmů

Proximální algoritmy jsou postupy z teorie optimalizace, které zahrnují celou škálu konvexních úloh, včetně relaxovaných úloh s ℓ_1 -normou. Tyto metody nacházejí v dnešní době poměrně široké uplatnění díky řídkým reprezentacím signálů. Samotný pojem proximální je synonymem slova blízký či blízký ke středu, což umožňuje konkrétnější představu o funkci těchto typů algoritmů.

Úlohy, na které lze použít proximální algoritmy spadají do kategorie metod založených na ℓ_1 relaxaci, označovaných jako relaxační algoritmy. Tyto algoritmy spo-
léhají na to, že za určitých podmínek (viz část 5.5) se lze dopracovat k přesnému nebo alespoň aproximovanému řešení. Fakt, že lze použít proximální algoritmy, je založen na skutečnosti, že ℓ_1 -norma je konvexní funkce [27].

7.2 Proximální operátory

Projekční operátor na konvexní množinu hledá bod v této množině ležící co nejbliž-
výchozímu bodu. Proximální operátor konvexní funkce je přirozeným rozšířením, jinak řečeno zobecněním, pojmu projekčního operátoru na konvexní množinu. Jeho užití poskytuje přínos v podobě možnosti práce s funkcemi, které nejsou diferencovatelné.

Obecně se dá říci, že tyto operátory mají zajímavé vlastnosti, které je činí vhod-
nými pro užití v iterativních minimalizačních algoritmech. Z perspektivy zpracování signálů mají velmi přirozenou interpretaci pro odšumování signálů [36].

Pro proximální operátor konvexní funkce f pro každé $\mathbf{x} \in \mathbb{R}^N$ značený $\text{prox}_f(\mathbf{x})$ platí, že

$$\arg \min_{\mathbf{y} \in \mathbb{R}^N} f(\mathbf{y}) + \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|_2^2. \quad (7.1)$$

Proximální operátor ℓ_1 -normy je funkce, která se nazývá *měkké prahování* (soft thresholding) a patří do třídy tzv. *shrinkage operátorů*. Má tvar

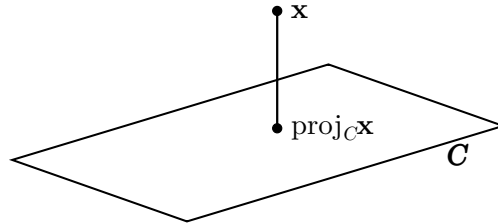
$$\text{soft}(\mathbf{x}) = \begin{cases} \mathbf{x} - \tau & \text{pro } \mathbf{x} < \tau, \\ 0 & \text{pro } \mathbf{x} \in \langle \tau, \bar{\tau} \rangle, \\ \mathbf{x} - \bar{\tau} & \text{pro } \mathbf{x} > \bar{\tau}, \end{cases} \quad (7.2)$$

kde parametr τ představuje práh.

Indikátorová funkce ι_C definovaná v [36] je užitečná funkce, která nabývá hodnoty ∞ , pokud \mathbf{x} leží mimo přípustnou množinu, a naopak pokud se nachází na dané množině, nabývá hodnoty 0. Funkci je možné zapsat jako

$$\iota_C(\mathbf{x}) = \begin{cases} 0 & \text{pro } \mathbf{x} \in C, \\ \infty & \text{pro } \mathbf{x} \notin C, \end{cases} \quad (7.3)$$

kde C je podmnožinou \mathbb{R}^N . Proximální operátor indikátorové funkce ι_C je projekce na množinu přípustných řešení C , značený jako $\text{proj}_C(\mathbf{x})$. Projekce na množinu C je v podstatě zobrazení, které přesune \mathbf{x} , které neleží na množině C , nejkratší cestou na bod na množině C , jak je pro názornost zobrazeno na obrázku 7.1.



Obr. 7.1: Znázornění projekce vektoru \mathbf{x} na množinu C .

7.3 Metody proximálního dělení

Mnoho problémů v oblasti zpracování signálů může být souhrně formulováno jako konvexní optimalizační úloha ve tvaru

$$\min_{\mathbf{x} \in \mathbb{R}^N} f_1(\mathbf{x}) + \dots + f_m(\mathbf{x}), \quad (7.4)$$

kde f_1, \dots, f_m jsou konvexní funkce z \mathbb{R}^N v rozsahu $(-\infty, +\infty)$.

Metody proximálního dělení postupují rozdělením funkcí f_1, \dots, f_m tak, že je každá zpracována jednotlivě pro dosažení snadno implementovatelného algoritmu. Nazývají se proximální, protože každá nehladká funkce ze vztahu (7.4) je zahrnuta prostřednictvím svého proximálního operátoru [36].

Aby bylo možné úlohu řešit pomocí proximálního operátoru, je potřeba ji nejprve převést na tzv. *neomezený* (unconstrained) tvar. Tzv. *omezený* (constrained) tvar funkce se nazývá proto, že funkce má své omezující podmínky. Tyto podmínky lze s využitím indikátorové funkce ι_C vyjádřit jako neomezený tvar pomocí minimalizace součtu m funkcí.

Funkce f_1 je konvexní funkce, která je diferencovatelná s β -Lipschitzovským spojitým gradientem ∇g , tj. pro každé $[\mathbf{x}, \mathbf{y}]$ z prostoru $\mathbb{R}^N \times \mathbb{R}^N$ platí

$$\|\nabla g(\mathbf{x}) - \nabla g(\mathbf{y})\| \leq \beta \|\mathbf{x} - \mathbf{y}\|, \quad (7.5)$$

kde β může nabývat hodnot v rozsahu $(0, \infty)$. Pokud funkce splňují výše uvedené podmínky, lze nalézt minimálně jedno fixní řešení pro $\gamma \in (0, \infty)$, které vystihuje rovnice

$$\mathbf{x} = \text{prox}_{\gamma f_1}(\mathbf{x} - \gamma \nabla f_2(\mathbf{x})), \quad (7.6)$$

kde pro $\mathbf{x} \in \mathbb{R}^N$ navíc platí ekvivalence se vztahem (7.4). Vzniká také možnost iterativního numerického řešení

$$\mathbf{x}_{n+1} = \underbrace{\text{prox}_{\gamma_n f_1}}_{\text{zpětný krok}} \left(\underbrace{\mathbf{x}_n - \gamma_n \nabla f_2(\mathbf{x}_n)}_{\text{dopředný krok}} \right), \quad (7.7)$$

které bude konvergovat k přesnému řešení. Index n v tomto případě zastupuje číslo iterace.

Tato metoda se nazývá *dopředně-zpětné dělení* (forward-backward splitting) nebo taktéž *proximální gradientní metoda* a je možné ji použít právě pro úlohu declippingu. Skládá se ze dvou kroků. Dopředného, gradientního kroku, ve kterém je využito pouze f_2 a zpětného, proximálního kroku, který využívá pouze f_1 [36].

7.4 Condatův algoritmus

Condatův algoritmus je obecný algoritmus pro konvexní optimalizaci, jehož autorem je Laurent Condat. Prakticky řeší úlohu

$$\arg \min_{\mathbf{x} \in \mathbb{C}^P} f(\mathbf{x}) + g(\mathbf{x}) + \sum_{m=1}^M h_m(L_m \mathbf{x}), \quad (7.8)$$

kde g je diferencovatelná v \mathbb{R}^N s β -Lipschitzovským spojitým gradientem ∇g podle vztahu (7.5).

Operátory L_m jsou linerární a ohraničené (*bounded*). Některé funkce vztahu (7.8) mohou být nulové a lineární operátor L_m může být identita [37]. Laurent Condat ve svém článku přichází se dvěma algoritmy, které řeší problém tzv. *plným rozdělením* (full splitting). To znamená, že v každé iteraci jsou jedinými prováděnými operacemi ∇g , prox_f , prox_{h_m} , L_m nebo hermitovsky sdružený operátor L_m^* . První z představených algoritmů je uvedený níže.

Algorithm 1: Condatův algoritmus

Výběr parametrů $\tau > 0, \sigma > 0, \rho > 0$.
 Počáteční odhady $\mathbf{x}^{(0)}, \mathbf{u}_1^{(0)}, \dots, \mathbf{u}_M^{(0)}$.

```

for  $i = 0, 1, \dots$  do
     $\tilde{\mathbf{x}}^{(i+1)} = \text{prox}_{\tau f}(\mathbf{x}^{(i)} - \tau \nabla g(\mathbf{x}^{(i)}) - \tau \sum_{m=1}^M L_m^* \mathbf{u}_m^{(i)})$ 
     $\mathbf{x}^{(i+1)} = \rho \tilde{\mathbf{x}}^{(i+1)} + (1 - \rho) \mathbf{x}^{(i)}$ 
    for  $m = 1, \dots, M$ , do
         $\tilde{\mathbf{u}}_m^{(i+1)} = \text{prox}_{\sigma h_m^*}(\mathbf{u}_m^{(i)} + \sigma L_m(2\tilde{\mathbf{x}}^{(i+1)} - \mathbf{x}^{(i)}))$ 
         $\mathbf{u}_m^{(i+1)} = \rho \tilde{\mathbf{u}}_m^{(i+1)} + (1 - \rho) \mathbf{u}_m^{(i)}$ 
    end
end
return  $\mathbf{x}^{(i+1)}$ 

```

Zde použité Fenchel-Rockafellarovo sdružení h_m označené jako h_m^* splňuje Moreovu identitu a pro reálné $\sigma > 0$ platí

$$\text{prox}_{\sigma h_m^*}(\mathbf{u}) = \mathbf{u} - \sigma \text{prox}_{h_m/\sigma}(\mathbf{u}/\sigma). \quad (7.9)$$

Autor v [37] předkládá podmínky pro konvergenci algoritmu při řešení problému, pokud parametry splňují

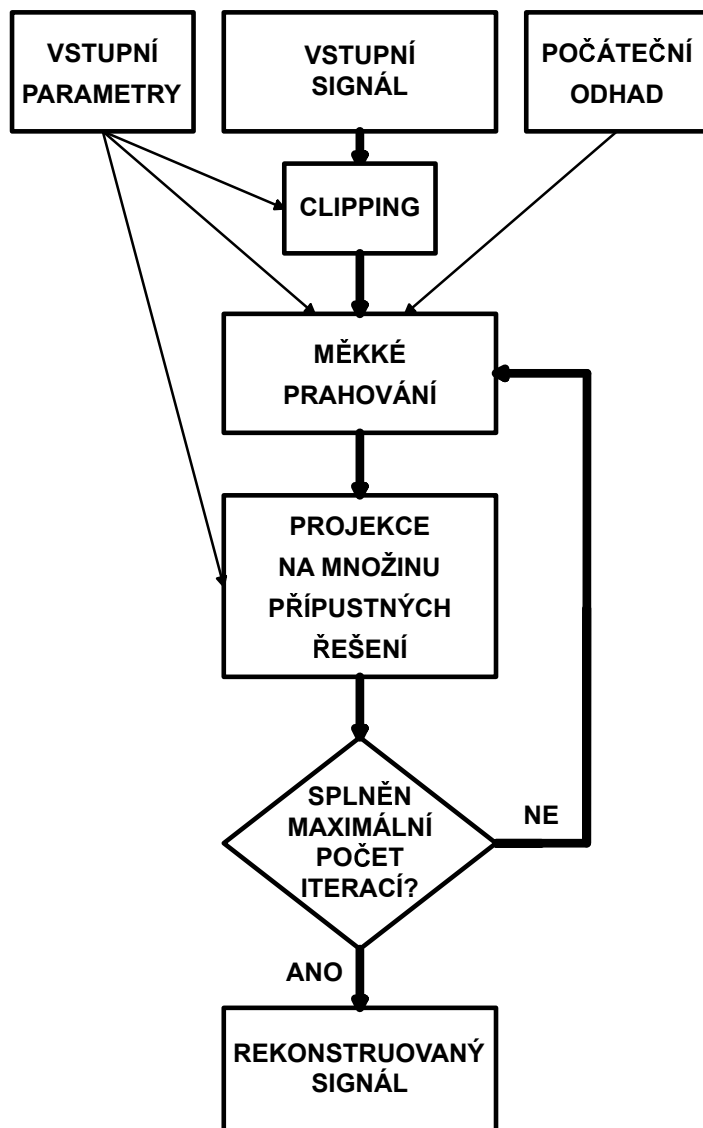
$$\tau \left(\frac{\beta}{2} + \sigma \left\| \sum_{m=1}^M L_m^* L_m \right\| \right) < 1, \quad (7.10)$$

$$\rho \in (0, 1). \quad (7.11)$$

Lipschitzova konstanta β byla definována v (7.5) a $\|\cdot\|$ představuje operátorovou normu. Dále autor uvádí, že pokud $g = 0$ a prostory funkcí jsou konečných rozměrů, algoritmus konverguje v případě splnění podmínek

$$\tau \sigma \left\| \sum_{m=1}^M L_m^* L_m \right\| \leq 1, \quad (7.12)$$

$$\rho \in (0, 2). \quad (7.13)$$



Obr. 7.2: Blokové schéma declippingu pomocí Condatova algoritmu.

7.5 Douglas-Rachfordův algoritmus

Dopředně-zpětný algoritmus uvedený v části 7.3 vyžaduje, aby jedna z funkcí byla diferencovatelná Lipschitzovským spojitým gradientem. V případě, že se odpustí tento předpoklad a nahradí se proximálním operátorem, jedná se o Douglas-Rachfordův algoritmus. Jeho řešení pro $\gamma \in (0, +\infty)$ je určeno touto dvou úrovnovou podmínkou uvedenou v [36]

$$\begin{cases} \mathbf{x} = \text{prox}_{\gamma f_2} \mathbf{y} \\ \text{prox}_{\gamma f_2} \mathbf{y} = \text{prox}_{\gamma f_1} (2\text{prox}_{\gamma f_2} \mathbf{y} - \mathbf{y}), \end{cases} \quad (7.14)$$

ze které je odvozen následující zápis algoritmu. Pro něj platí, že každou iterací \mathbf{x}_n , kde $n \in \mathbb{N}$, konverguje k řešení problému (7.4).

Algorithm 2: Douglas-Rachfordův algoritmus

Výběr parametrů $\epsilon \in (0, 1), \gamma > 0, \mathbf{y}_0 \in \mathbb{C}^P$.

```
for  $n = 0, 1, \dots$  do
     $\mathbf{x}_n = \text{prox}_\gamma f_2 \mathbf{y}_n$ 
     $\lambda_n \in [\epsilon, 2 - \epsilon]$ 
     $\mathbf{y}_{n+1} = \mathbf{y}_n + \lambda_n(\text{prox}_{\gamma f_1}(2\mathbf{x}_n - \mathbf{y}_n) - \mathbf{x}_n)$ 
end
return  $\mathbf{y}_{n+1}$ 
```

Douglas-Rachfordův algoritmus využívá stejně jako dopředně-zpětný algoritmus dělení, tj. pracuje s funkcemi f_1 a f_2 odděleně. To vede k myšlence, že by měl být ještě obecnější, protože nevyžaduje, aby nějaká funkce byla diferencovatelná Lipschitzovským spojitým gradientem. Na druhou stranu potřebuje poměrně numericky náročný výpočet, protože používá dva proximální operátory v každém kroku iterace, kdežto dopředně-zpětný algoritmus pouze jeden [38].

7.6 Chambolle-Pockův algoritmus

Tento algoritmus, jehož autory jsou Antonin Chambolle a Thomas Pock, je tzv. *primal-dual* metoda. To je metoda, která poskytuje v závislosti na primárním (*primal*) řešení také řešení duálního (*dual*) konvexního optimalizačního problému [40].

$$\min_{\mathbf{x} \in \mathbb{C}^P} f_1(K\mathbf{x}) + f_2(\mathbf{x}), \quad (7.15)$$

kde, K je spojitý lineární operátor. Funkce f_1 a f_2 jsou konvexní a nabývají rozsahu $(0, +\infty)$. Algoritmus má potom následující tvar, kde výrazy ∂f_1^* a ∂f_2 představují subgradienty konvexních funkcí f_1^* resp. f_2 a Id je identita [39]:

Algorithm 3: Chambolle-Pockův algoritmus

Výběr parametrů $\tau > 0, \sigma > 0, \theta \in [0, 1]$.

```
for  $n = 0, 1, \dots$  do
     $\mathbf{y}_{n+1} = (Id + \sigma \partial f_1^*)^{-1}(\mathbf{y}_n + \sigma K \bar{\mathbf{x}}_n)$ 
     $\mathbf{x}_{n+1} = (Id + \tau \partial f_2)^{-1}(\mathbf{x}_n - \tau K^* \mathbf{y}_{n+1})$ 
     $\bar{\mathbf{x}}_{n+1} = \mathbf{x}_{n+1} + \theta(\mathbf{x}_{n+1} - \mathbf{x}_n)$ 
end
return  $\bar{\mathbf{x}}_{n+1}$ 
```

8 Řešení úlohy declippingu

V této kapitole bude nejprve rozebrán obecný pohled na problematiku declippingu zvukového signálu, určena specifika pro tento problém a provedena obecná matematická formulace této minimalizační úlohy. V další části je zaměřeno již na konkrétní formu řešení tohoto problému. Jedná se o matematickou formulaci a návrh řešení pro syntetizující model declipperu za pomoci Condatova algoritmu a následně řešení, které v sobě zahrnuje psychoakustický model.

8.1 Obecná formulace

Pro úspěšné vyřešení úlohy se dopředu počítá s jistými předpoklady. Tím prvním je, že vstupní signál \mathbf{y} , který je oříznutý hard clippingem, je oříznutý symetricky, tzn. že dolní a horní prahová úroveň θ_c se liší pouze ve svém znaménku. Hard clipping byl proveden podle vztahu (1.1) a jeho výstupní signál je označený jako \mathbf{y}_c . Dále se předpokládá znalost indexů vzorků signálu, které byly oříznuty a to, že vzorky které oříznuty nebyly, přesně odpovídají vzorkům původního signálu.

Díky znalosti indexů ořezaných vzorků lze určit tzv. *masky* clippovaného signálu. Masky jsou v tomto případě vektory logických hodnot o délce vstupního signálu. V tomto případě je vhodné použít tři různé masky pro jeden signál. První maska odpovídá vzorkům, které nejsou ořezány. Tyto vzorky se označují jako spolehlivé (*reliable*) a označení pro tuto masku je \mathbf{M}_r . Vzorkům, které byly ořezány zespoda a jejich hodnota je přichycena na úroveň $-\theta_c$ odpovídá maska \mathbf{M}_l (*low*). A naopak, pro vzorky přichycené na horní úrovni odpovídá maska \mathbf{M}_h (*high*).

Pro restaurovaný signál platí, že hodnoty, které byly přichyceny na $-\theta_c$ se nyní musí nacházet pod touto úrovní a vzorky přichycené na $+\theta_c$ naopak nad touto prahovou úrovní. Dalším z předpokladů, které obecně splňuje většina zvukových signálů je to, že jsou řídké ve vhodné časově-kmitočtové reprezentaci [27]. Zde se předpokládá využití Diskrétní Gaborovy transformace a Diskrétní Fourierovy transformace.

Úloha restaurace clippovaného signálu tedy spočívá v nalezení nejřidšího vektoru koeficientů \mathbf{z} získaných pomocí DGT případně DFT, které zároveň splňují omezující podmínky. První člen omezující podmínky navíc dovoluje jistou odchylku na pozicích spolehlivých vzorků. Obecně se tedy dá úloha declippingu matematicky zapsat jako

$$\mathbf{x} = \arg \min_{\mathbf{z} \in \mathbb{C}^P} \|\mathbf{z}\|_0 \quad \text{vzhledem k} \quad \begin{cases} \|\mathbf{M}_r \mathbf{y} - \mathbf{M}_r D \mathbf{z}\|_2^2 \leq \epsilon \\ \mathbf{M}_h D \mathbf{z} \geq +\theta_c, \\ \mathbf{M}_l D \mathbf{z} \leq -\theta_c, \end{cases} \quad (8.1)$$

kde, pro parametr ϵ platí, $\epsilon > 0$ [29]. V tomto případě $D : \mathbb{C}^P \rightarrow \mathbb{R}^N$ značí syntetizující operátor (IDFT) a $\mathbf{x} \in \mathbb{C}^P$ jsou koeficienty, ze kterých získáme restaurovaný signál pomocí zpětné transformace.

8.2 Konkrétní řešení pomocí Condatova algoritmu

Pro řešení zadané úlohy byl zvolen Condatův algoritmus, který je rovněž znám pod označením *Obecný proximální algoritmus pro konvexní optimalizaci*. Úloha je zadaná třemi konvexními funkcemi a proto zůstává jedinou volbou, protože Douglas-Rachford a Chambolle-Pock algoritmy jsou definované pouze pro součet dvou konvexních funkcí.

V první části je provedena matematická formulace přímo pro Condatův algoritmus, dále nastavení parametrů vzhledem ke konvergenčním podmínkám a posléze rozebrán samotný algoritmus řešící zadanou úlohu. Další část navazuje na předchozí s tím rozdílem, že nyní je již aplikován psychoakustický model. Je zde objasněn postup pro výpočet globálního maskovacího prahu a zahrnutí váhovacího vektoru do výpočtu kmitočtově závislého měkkého prahování.

8.2.1 Formulace Condatova algoritmu

Jak již bylo zmíněno v části 7.3, aby bylo možné použít proximální operátory pro řešení optimalizační úlohy, je třeba převést omezený tvar optimalizačního problému

$$\mathbf{x} = \arg \min_{\mathbf{z}} \|\mathbf{z}\|_1 + \frac{1}{2\gamma} \|\mathbf{M}_r \mathbf{y} - \mathbf{M}_r D\mathbf{z}\|_2^2 \quad \text{vzhledem k} \quad \begin{cases} \mathbf{M}_h D\mathbf{z} \geq +\theta_c, \\ \mathbf{M}_l D\mathbf{z} \leq -\theta_c, \end{cases} \quad (8.2)$$

na neomezený tak, že se podmínka z pravé strany vyjádří jako součet m konvexních funkcí pomocí indikátorových funkcí, jejichž definice je v části 7.2. Potom bude mít úloha tvar

$$\mathbf{x} = \arg \min_{\mathbf{z}} \|\mathbf{z}\|_1 + \frac{1}{2\gamma} \|\mathbf{M}_r \mathbf{y} - \mathbf{M}_r D\mathbf{z}\|_2^2 + \iota_{hl}(D\mathbf{z}). \quad (8.3)$$

V optimalizační úloze (8.2) se oproti obecné úloze declippingu (8.1) přesunula podmínka pro vzorky na spolehlivých pozicích, která je nyní taktéž minimalizovaná a umožňuje během výpočtu jistou odchylku. Velikost této odchylky lze měnit pomocí penalizačního parametru γ . Vzhledem k tomu, že se řešení ve většině případů shoduje, byla pro hledání řídkého řešení nahrazena nekonvexní ℓ_0 -norma z úlohy (8.1) konvexní ℓ_1 -normou, jak bylo uvedeno v části 5.5. Množina přípustných řešení je v tomto případě vyjádřena pomocí sjednocení dvou množin. Pro množinu h a l platí $h = \{D\mathbf{z} | \mathbf{M}_h D\mathbf{z} \geq +\theta_c\}$ resp. $l = \{D\mathbf{z} | \mathbf{M}_l D\mathbf{z} \leq -\theta_c\}$. Zápis pomocí

jedné množiny by potom vypadal následovně $hl = \{D\mathbf{z} | \mathbf{b}_l \leq D\mathbf{z} \leq \mathbf{b}_h\}$. Pro vektory \mathbf{b}_l a \mathbf{b}_h platí

$$\mathbf{b}_l = \begin{cases} -1 & \text{pro indexy v } \mathbf{M}_l, \\ +\theta_c & \text{pro indexy v } \mathbf{M}_h, \end{cases} \quad (8.4)$$

$$\mathbf{b}_h = \begin{cases} -\theta_c & \text{pro indexy v } \mathbf{M}_l, \\ +1 & \text{pro indexy v } \mathbf{M}_h. \end{cases} \quad (8.5)$$

Tvar úlohy (8.3) již koresponduje s podobou úlohy Condatova algoritmu (7.8). V tomto případě nebude ani jedna z funkcí nulová a přestávají tak platit konvergenční podmínky (7.12) a (7.13), které předpokládají $g = 0$.

Funkce f odpovídá ℓ_1 -normě vektoru a pro její proximální operátor platí

$$\text{prox}\|\mathbf{z}\|_1 = \text{soft}_{\tau z}(\text{sgn}(\mathbf{z}_n) \max\{|\mathbf{z}_n| - \tau, 0\})_{n=1}^N. \quad (8.6)$$

Pro diferencovatelnou funkci g existuje kromě proximálního operátoru také její gradient, který má po dosazení podobu

$$\nabla \frac{1}{2\gamma} \|\mathbf{M}_r \mathbf{y} - \mathbf{M}_r D\mathbf{z}\|_2^2 = \frac{1}{\gamma} \mathbf{M}_r D^* (D\mathbf{z} - \mathbf{y}). \quad (8.7)$$

Operátor D je syntetizující operátor lineární transformace a D^* analyzující operátor. Tyto operátory korespondují se značením operátorů L_m a L_m^* ze vztahu (7.8). Sumě funkcí h_m odpovídá jediná indikátorové funkci $\iota_{hl}(D\mathbf{z})$ a její proximální operátor je projekce na množinu hl .

8.2.2 Nastavení parametrů

Konvergenční podmínku (7.10) lze následovně zjednodušit. Pro tento případ platí, že Lipschitzova konstanta β bude rovna jedné. Operátorová norma $\|D^*D\|$ bude rovna taktéž 1, protože jsou použity Parsevalovy těsné framy (viz část 6.3). Tímto zjednodušením podmínky nyní vyplývá

$$\tau\left(\frac{1}{2} + \sigma\right) < 1 \quad \text{respektive} \quad \tau < \frac{2}{1 + 2\sigma}. \quad (8.8)$$

Parametry τ a σ slouží v algoritmu k váhování mezi měkkým prahováním a projekcí na množinu a jejich vzájemná závislost tvoří hyperbolu.

Podmínka (7.11) stanovuje, že by se parametr ρ měl pohybovat v rozmezí $(0, 1)$. Interval je zleva otevřený, protože při nastavení $\rho = 0$ by nedocházelo k posunu v hledání řešení.

Pro nastavení penalizačního parametru γ platí obecná podmínka $\gamma > 0$, přestože konvergence algoritmu závisí na vhodné kombinaci s ostatními parametry. Bez znalosti parametrů σ a ρ tedy nelze dopředu určit optimální interval, ve kterém by se měl parametr γ nacházet.

8.2.3 Algoritmus a jeho popis

Na dalších řádcích je specifická podoba algoritmu pro řešení zadané úlohy odvozená z algoritmu uvedeného v části 7.4.

Algorithm 4: Condatův algoritmus pro řešení (8.3)

Výběr parametrů $\gamma > 0, \sigma > 0,5, \tau < \frac{2}{1+2\sigma}, \rho \in (0, 1)$.

Počáteční odhady $\mathbf{z}^{(0)}, \mathbf{u}_{hl}^{(0)}, \mathbf{y}^{(0)}$.

for $i = 0, 1, \dots$ **do**

$$\tilde{\mathbf{z}}^{(i+1)} = \text{soft}_{\tau} \left(\mathbf{z}^{(i)} - \tau \frac{1}{\gamma} \mathbf{M}_r D^* (D\mathbf{z}^{(i)} - \mathbf{y}) - \tau D^* \mathbf{u}_{hl}^{(i)} \right)$$

$$\mathbf{z}^{(i+1)} = \rho \tilde{\mathbf{z}}^{(i+1)} + (1 - \rho) \mathbf{z}^{(i)}$$

$$\mu = \mathbf{u}_{hl}^{(i)} + \sigma D(2\mathbf{z}^{(i+1)} - \mathbf{z}^{(i)})$$

$$\tilde{\mathbf{u}}_{hl}^{(i+1)} = \mu - \sigma \text{proj}_{hl} \left(\frac{\mu}{\sigma} \right)$$

$$\mathbf{u}_{hl}^{(i+1)} = \rho \tilde{\mathbf{u}}_{hl}^{(i+1)} + (1 - \rho) \mathbf{u}_{hl}^{(i)}$$

end

return $\mathbf{z}^{(i+1)}$

Vstupní parametry a jejich nastavení jsou už více specifikovány pro tuto úlohu. Jako počáteční odhady je zde použitý vstupní clippovaný signál a jeho kmitočtové koeficienty získané analýzou signálu. Současný algoritmus obsahuje pouze jednu smyčku a to díky tomu, že se podařilo vyjádřit přípustná řešení pomocí jediné množiny hl . Obecné značení proximálních operátorů již bylo nahrazeno konkrétními operátory. V prvním kroku algoritmu je to měkké prahování označené jako *soft* a ve čtvrtém kroku pak projekce na množinu hl označená jako *proj_{hl}*. Ve třetím kroku se nachází pomocný výpočet μ .

Na analyzující operátor D^* byla aplikována maska \mathbf{M}_r , která zajistí, že se analýza rozdílu $(D\mathbf{z}^{(i)} - \mathbf{y})$ bude týkat jen signálu na spolehlivých pozicích. Proximální operátor, kde se nacházelo Fenchel-Rockafellarovo sdružení h_m označené jako h_m^* je zde už upraveno podle vztahu (7.9).

8.3 Řešení s využitím psychoakustického modelu

Jak již bylo zmíněno v kapitole 3, zahrnutí psychoakustického modelu do algoritmu pro declipping umožňuje při restauraci signálu zaměření se na ty kmitočtové složky, které jsou percepčně významné, tj. mají velký vliv na kvalitu a naopak zanedbání složek, které prakticky člověk neslyší, jelikož se nachází pod maskovacím prahem.

8.3.1 Perceptuální váhovací funkce

Aby objektivní funkce $\|\mathbf{z}\|_1$ v optimalizačním problému správně reflektovala množství vnímatelného zkreslení, které má na určitých kmitočtech větší dopad než na jiných, je třeba zahrnout do této funkce *kmitočtově závislé váhování*. Kombinací absolutního prahu slyšení (viz část 3.1.1) a kmitočtového maskování (viz část 3.1.2) lze získat hodnotu energie zkreslení pro každý kmitočet, který může být maskován [45]. Výpočet globální maskovacího prahu GMT (*global masking threshold*) využívá první vrstvu psychoakustického modelu MPEG 1. Celý postup se dohromady skládá z pěti kroků:

1. Spektrální analýza a normalizace SPL.
2. Identifikace tónových a šumových složek maskerů.
3. Decimace maskerů.
4. Výpočet jednotlivých maskovacích prahů pro každý masker.
5. Výpočet globálního maskovacího prahu.

Zahrnutí kmitočtově závislého váhování do výpočtu je provedeno pomocí perceptuálního váhovacího vektoru \mathbf{w} . Získaný vektor \mathbf{w} je posléze doplněn do vztahu (8.3), který má nyní tuto podobu:

$$\mathbf{x} = \arg \min_{\mathbf{z}} \|\mathbf{z} \odot \mathbf{w}\|_1 + \frac{1}{2\gamma} \|\mathbf{M}_r \mathbf{y} - \mathbf{M}_r D\mathbf{z}\|_2^2 + \iota_{hl}(D\mathbf{z}). \quad (8.9)$$

Symbol \odot zde značí Hadamardův součin, resp. násobení matic či vektorů po složkách. Vzhledem k tomu, že objektivní funkce nyní obsahuje i váhovací vektor \mathbf{w} , je nadále přesnější hovořit o váhované ℓ_1 -normě.

Váhovací vektor $\mathbf{w} \in \mathbb{R}^P$ obsahuje jednotlivé inverzní a normalizované hodnoty maskovacích prahů v decibelech, které indikují relativní perceptuální důležitost různých kmitočtových složek signálu, a je definován jako

$$\mathbf{w} = (\mathbf{gmt}_1^{-1}, \mathbf{gmt}_2^{-1}, \dots, \mathbf{gmt}_P^{-1}), \quad (8.10)$$

kde \mathbf{gmt}_p označuje p -tý segment, pro který byl GMT vypočítán. V každém vektoru \mathbf{gmt}_p jsou tedy obsaženy jednotlivé kmitočtové složky globálního maskovacího prahu daného segmentu. Ještě před použitím váhovacího vektoru \mathbf{w} dochází k omezení dynamického rozsahu. Omezení se používá proto, že křivka absolutního prahu slyšení je definována od 0 Hz do $F_s/2$, kde F_s je vzorkovací kmitočet, a to znamená, že v krajních hodnotách dochází k velkému nárůstu úrovně prahu slyšení. To ale není třeba, a tak po omezení dynamického rozsahu se hodnoty překračující nastavenou mez „zarovnají“ na tuto hodnotu. Po omezení dynamického rozsahu dochází k inverzi podle vztahu

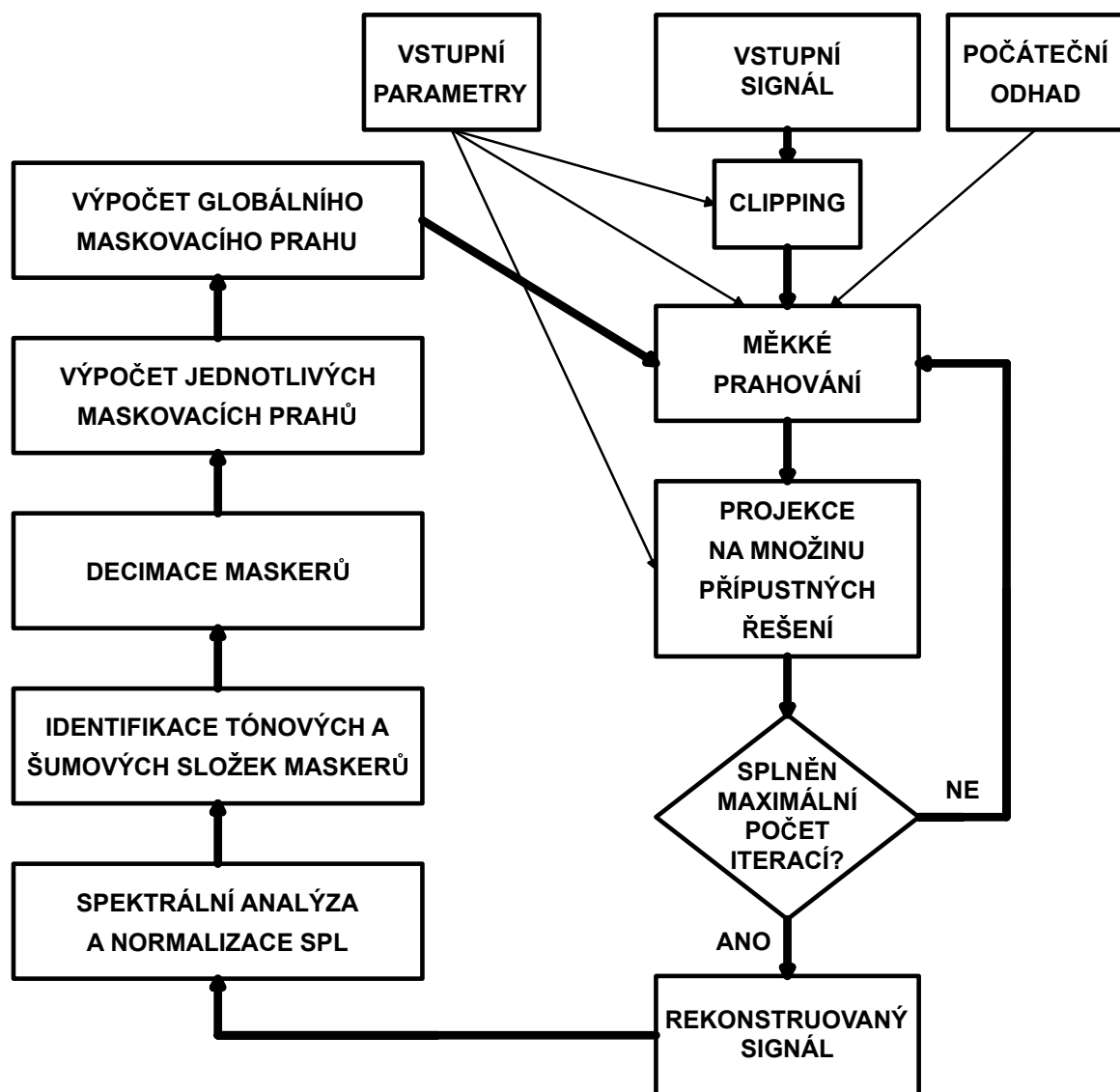
$$\mathbf{w} = \left((\mathbf{gmt}_p - \min(\mathbf{gmt}_p) + 1)^{-1} \right)_{p=1}^P. \quad (8.11)$$

Posledním krokem je normalizace, která má kladný vliv na rychlost konvergence algoritmu. Ovlivňuje také nutnost přenastavení parametrů algoritmu pro získání uspokojivých výsledků.

Tímto způsobem pak váhovací vektor upřednostňuje kmitočtové složky s vysokým maskovacím prahem. Naopak zkreslením nově vzniklé výrazné složky na vysokých kmitočtech s nízkým maskovacím prahem, které nebyly součástí původního signálu jsou potlačeny. Nově vzniklé složky s vyšším maskovacím prahem, které jsou méně slyšitelné, jsou nyní tolerovány ve větší míře [29].

Ve schématu s obecným Condatovým algoritmem (viz obrázek 7.2) na následující straně byl do zpětné vazby přidán psychoakustický model, jehož vstupem je již jednou rekonstruovaný signál. Po vytvoření globálního maskovacího prahu je doplněn váhovací vektor \mathbf{w} do objektivní funkce a rekonstrukce začíná znovu, tentokrát již s váhováním koeficientů.

Zde je vhodné upozornit, že tento postup se aplikuje, pokud nedochází k segmentaci signálu. Při použití segmentace probíhá rekonstrukce pouze jednou a to tak, že první segment je rekonstruovaný bez použití váhovacího vektoru. Ten se vypočítá následně z prvního rekonstruovaného segmentu a použije se pro váhování druhého segmentu atd. Vzhledem k délce segmentů by měl být rozdíl globálních maskovacích prahů mezi jednotlivými segmenty ve většině případů minimální.



Obr. 8.1: Blokové schéma declippingu pomocí Condatova algoritmu s využitím psychoakustického modelu.

9 Programové řešení úlohy

Náplní této kapitoly je popis konkrétního programového řešení declippingu implementovaného v prostředí *MathWorks Matlab verze R2016a 9.0*. Řešení se opírá především o poznatky z části 7.4 a kapitoly 8. Aby bylo možné program spustit, je třeba mít nainstalovaný toolbox LTFAT pro práci s časově-kmitočtovou analýzou a syntézou, který je volně dostupný na internetu [48].

Nejprve je nastíněna struktura kódu a nástin principu činnosti. Dále je detailně popsán obsah a funkčnost jednotlivých hlavních souborů a jejich funkcí.

9.1 Struktura kódu

Kód je strukturován tak, že byly vytvořeny tři hlavní soubory, ze kterých lze spustit rekonstrukci signálu. Další soubory obsahují pomocné funkce, které jsou volány z hlavních souborů případně v rámci dalších funkcí.

První soubor `main_basic.m` obsahuje implementaci Condatova algoritmu. Druhý soubor `main_psyac.m` obsahuje tentýž algoritmus, ale navíc je zde implementovaný psychoakustický model. Třetí a poslední hlavní soubor `main_psyac_segment.m` obsahuje Condatův algoritmus včetně psychoakustického modelu, ovšem s tím rozdílem, že nyní je signál zpracováván po jednotlivých segmentech.

Všechny tři hlavní soubory mají v základu stejný princip činnosti. Před spuštěním výpočtu je třeba nastavit parametry a volitelné zobrazení výsledků či zápis do výstupního audio souboru. Po načtení vstupního signálu proběhne hard clipping. Před voláním samotného Condatova algoritmu probíhají pomocné výpočty a matricové úpravy. Algoritmus vrátí rekonstruované koeficienty, které jsou převedeny zpět do časové oblasti. Z toho signálu jsou poté spočítané hodnoty ΔSDR a případně zobrazen vizuální výstup pro bezprostřední možnost vyhodnocení. Nakonci proběhne výpis údajů z rekonstrukce do konzole.

9.2 Hlavní soubor 1

Hlavní soubor 1 je označen pro m-soubor s názvem `main_basic.m`. Ten provádí declipping pomocí Condatova algoritmu.

Na samém začátku skriptu je volán příkaz `ltfatstart`, který spustí LTFAT Toolbox. Stačí, aby tento příkaz proběhl jednou při prvním spuštění. Před spuštěním je nutné nastavit vlastní cestu k adresáři s LTFAT.

Před samotným nastavením parametrů algoritmu je možnost pro vygenerování zvukového nebo grafických výstupů pomocí zadání logických hodnot. Popis jednotlivých funkcí je součástí jednoho z následujících pododdílů.

9.2.1 Parametry

V této části probíhá nastavení jednotlivých parametrů, které jsou uloženy do společné struktury s názvem `par` z důvodu jejich využití v rámci některých funkcí. Při volání funkce pak není třeba definovat jednotlivé vstupní parametry, namísto toho je použita struktura zahrnující tyto parametry a v dané funkci pak s nimi lze pracovat pomocí tečkové konvence.

První část kódu obsahuje nastavení jednotlivých parametrů algoritmu `theta`, `gamma`, `sigma`, `rho` a `tau`, které jsou objasněny v předchozí kapitole. Parametr `inEq` slouží vytvoření nerovnosti, aby byla dodržena platnost vztahu (8.8).

V další části kódu je nastavení pro maximální počet iterací algoritmu a terminační kritérium. Oba parametry při splnění podmínky mohou nezávisle na sobě ukončit program. Při testování této varianty bylo obvyklé, že relativní odchylka byla mezi vstupními a rekonstruovanými koeficienty nižší než 10^{-5} dříve, než došlo k překročení maximálního počtu iterací. Tato hodnota byla v tomto případě nastavena na 2000.

Poslední část kódu se týká parametrů použitého framu. V tomto případě byl použit frame `DGTreal`, který vrací Gaborovy koeficienty pouze pro kladné kmitočty. Jako váhovací okno bylo zvoleno Hannovo okno, se kterým bylo dosaženo nejlepších výsledků, ale v případě potřeby jej lze změnit. Dále je možnost nastavení velikosti a posuvu (překryvu) okna a počtu kmitočtových kanálů, jejichž počet v tomto případě odpovídá velikosti okna. Překryv okna je nastaven na polovinu délky okna, tedy na 50 %.

9.2.2 Načtení vstupního signálu

Vstupní signál je načten pomocí dialogového okna, které umožňuje pohodlné vyhledání požadovaného audio souboru v adresářích počítače. Dialogové okno se automaticky otevře po spuštění skriptu. Samotné načtení probíhá pomocí funkce `audioread`. Soubor je uložený do proměnné `INPUT_signal` a kromě samotného souboru ve formátu WAV je získán i vzorkovací kmitočet `Fs`. V dalším kroku probíhá normalizace tohoto signálu.

9.2.3 Clipping

Z hlavního souboru je pro hard clipping vstupního signálu volána funkce `hard_clip`. Ta na svém začátku obsahuje ošetření proti nevhodně nastavenému parametru `theta`, který by se měl pohybovat v rozsahu hodnot 0 až 1.

Vstupními parametry této funkce je vstupní signál `INPUT_signal` a parametr `theta`, který určuje prahovou úroveň, na které dojde k symetrickému oříznutí signálu. Výstupem jsou kromě clippovaného signálu označeného jako `CLIP_signal` také masky `Mr`, `Mh` a `Ml` uložené do struktury `masks`.

9.2.4 Příprava před zpracováním

V této části kódu probíhá nejprve konstrukce DGTreal framu pomocí LTFAT funkce `frame`. Je zde vytvořena také čtvrtá maska označená jako `Mh1`, která představuje sjednocení masek `Mh` a `Ml` a slouží k získání signálu pouze na pozicích, které byly clippované. Dále probíhá úprava délky signálu a masek, aby odpovídaly násobkům velikosti okna. Po této úpravě má `CLIP_signal` a struktura `masks` označení s přídomkem `extended`.

9.2.5 Algoritmus declippingu

Nejdříve jsou získány koeficienty `z` pomocí funkce `frana`, kde jsou vstupními parametry zkonstruovaný frame `F` a `CLIP_signal`. Proměnná `u_h1` obsahuje signál `CLIP_signal_extended`, na který byla aplikována maska `Mh1`. Proměnná `y` pak obsahuje celý signál `CLIP_signal_extended`.

Tyto proměnné posléze vstupují jako vstupní argumenty do funkce `condat.m`, která bude podrobněji popsána dále.

Výstupem jsou rekonstruované koeficienty `x`, které jsou pomocí funkce `frsyn` převedeny zpět do časové oblasti. Dalšími výstupy jsou informace u době rekonstrukce získané pomocí příkazů Matlabu `tic` a `toc` a informace o počtu iterací potřebných ke konvergenci algoritmu. Na závěr je vypočtena hodnota objektivní funkce, tedy ℓ_1 -norma koeficientů `x`.

9.2.6 Výsledky rekonstrukce a jejich zobrazení

Výsledky rekonstrukce jsou vyhodnoceny pomocí objektivní metody testování kvality zvukových signálů, v tomto případě zjištěním rozdílu SDR (viz kapitola 4). K tomu slouží funkce `SDR_calc`, která podle vztahu (4.2) vypočítá odstup výkonu vstupního signálu od výkonu zpracovaného signálu. Buď clippovaného nebo rekonstruovaného. K dispozici je informace ΔSDR , která informuje o rozdílu mezi SDR vstupního a rekonstruovaného a vstupního a clippovaného. Druhá informace ΔSDR_{RR} je rozdílná v tom, že v rekonstruovaném signálu byly nahrazeny vzorky na spolehlivých pozicích pomocí funkce `reliable_replace`. To se provádí z toho důvodu, že při hledání řešení je dovolena odchylka na spolehlivých pozicích a v některých

případech dochází ke značné degradaci signálu. Porovnáním těchto dvou hodnot lze například zjistit, k jak velkému odchýlení došlo.

V případě nastavení pro vygenerování zvukového a vizuálních výstupů rekonstrukce dochází k volání funkce `wav_generator`, která z vektoru rekonstruovaného signálu vytvoří soubor ve formátu WAV, jehož název obsahuje i hodnoty vstupních parametrů předaných pomocí struktury `string`.

Funkce `visualization_spectrogram` poskytuje zobrazení tří spektrogramů (viz část 1.4). Funkce nemá nadefinované žádné výstupní parametry, pouze na monitor vedle sebe zobrazí tři spektrogramy. Vlevo spektrogram vstupního signálu, uprostřed clippovaného a vpravo rekonstruovaný signál. Lze tak rychle získat přibližné povědomí o tom, jak úspěšná byla rekonstrukce.

Jako třetí funkce zde figuruje `visualization_waveform`, což je zobrazení časového průběhu signálu. V jednom grafu jsou umístěny průběhy vstupního, clippovaného i rekonstruovaného signálu s nahrazenými vzorky na spolehlivých pozicích.

9.3 Hlavní soubor 2

V tomto oddílu bude popsán soubor `main_psyac.m`. Je shodný s `main_basic.m`, ale obsahuje navíc implementaci psychoakustického modelu.

Dále budou tedy rozebrány pouze odlišnosti. První odlišností jsou přidáné vstupní parametry algoritmu. Parametr `weighting` může provádět „bypass“ psychoakustického modelu, model umožňuje vybrat tzv. *spreading function*, která definuje tvar maskovací křivky pro čistý tón. Jako výchozí hodnota je použitý model *MPEG 1*, ale lze zvolit i *MPEG 2*, *modifikovanou Schröderovu funkci* či prosté přímky připomínající trojúhelník bez základny označený jako *2 – slope*. Následujícím parametrem je `dyn_range`, který umožňuje nastavit dynamický rozsah GMT (viz část 8.3). Parametr `masking_win` umožňuje použít či obejít váhování signálu oknem před výpočtem GMT. Je to z toho důvodu, že funkci pro vytvoření GMT používá i varianta se segmentací signálu, kde již váhování oknem probíhá a váhování dvakrát by bylo zbytečné. Další odlišností je rozdílné nastavení maximálního počtu iterací, protože bylo empiricky zjištěno, že rekonstrukce s použitím psychoakustického modelu vyžaduje více iterací ke konvergenci.

9.3.1 Psychoakustický model

Aby bylo zajištěno, že globální maskovací práh bude získaný z rekonstruovaného signálu, je třeba provést rekonstrukci dvakrát. Poprvé bez a podruhé už s psychoakustickým modelem. To je prakticky vyřešeno vytvořením další smyčky pouze o dvou iteracích, kde se podmínkou v první iteraci obejde volání funkce pro vytvoření GMT.

Před výpočtem GMT je potřeba upravit podobu vektoru signálu z prvotní rekonstrukce do podoby matice, kde délka sloupců odpovídá délce okna DGT framu. To zajišťuje funkce `weighting_init`.

Globální maskovací práh je získán pomocí funkce `masking`, která byla převzata a upravena se svolením autora, kterým je docent Jiří Schimmel. Vstupními parametry této funkce jsou rekonstruovaný signál `REC_signal_temp`, resp. segment signálu o velikosti okna a struktura `par`. Výstupem této funkce je vektor `GMT`, který obsahuje normalizovanou inverzní maskovací křivku. Jednotlivé vektory `GMT` jsou poté poskládány do vektoru `w`, jehož délka odpovídá délce vektoru `s` koeficienty. Tento vektor se nazývá váhovací vektor.

Poté, co se získá vektor `wah`, je proveden součin s parametrem `tau`. Tento parametr doposud sloužil jako práh měkkého prahování. Nyní tedy v rámci Condatova algoritmu nedochází k prahování skalární veličinou, nýbrž celým vektorem. Počet prvků vektoru `w` odpovídá počtu koeficientů uložených v proměnné `z`.

9.4 Hlavní soubor 3

Hlavní soubor 3 odpovídá m-souboru `main_psyac_segment.m`, který při declippingu používá segmentaci signálu. Kód realizující segmentaci signálu byl použit se svolením autora Ing. Pavla Závišky. V tomto případě je použita DFT transformace, přesněji `dftreal`. Rozdíl je nyní v tom, že je potřeba nastavit překryv okna na 75 %, protože použití pouze 50% překryvu má velmi negativní vliv na výsledek rekonstrukce narozdíl od předchozích dvou variant.

V části kódu nazvané „Příprava před zpracováním“ je nyní navíc vytvoření analyzujícího a syntetizujícího okna, příprava indexace segmentů, která nahrazuje zabudovanou funkci Matlabu `fftshift`. Probíhá zde také inicializace vektorů pro segmentové zpracování. Postupné rekonstruování jednotlivých segmentů probíhá ve smyčce `for`, kde jsou nejdříve upraveny indexy segmentů vstupujících do zpracování Condatovým algoritmem a také jejich násobení analyzujícím oknem `gana`. To se týká segmentu clippovaného signálu, masek, ale také vektoru, který obsahuje hodnoty parametru `theta` násobené váhovacím oknem. Před rekonstrukcí je podmínka, která určí, že první segment bude prahován pouze parametrem `tau`. Po rekonstrukci a syntéze koeficientů dochází k volání funkce `ifftshift` a následně jsou do proměnné `REC_signal` s rekonstruovaným signálem skládány za sebe jednotlivé segmenty pro násobení syntetizujícím oknem `gsyn`.

Následně probíhá volání funkce `masking`, do které vstupuje rekonstruovaný segment. Jak již bylo zmíněno, psychoakustický model je implementovaný ve zpětné vazbě. To znamená, že získaný vektor váh `w` je použitý při rekonstrukci následujícího segmentu.

9.5 Condatův algoritmus

V této funkci s názvem `condat.m` je implementovaný Obecný proximální algoritmus odvozený v části 8.2. Tuto variantu volá `main_basic.m` a `main_psyac.m`.

Vstupními parametry této funkce jsou koeficienty `z`, signál `u_hl` a `y`, struktura obsahující masky a parametry, vektor váh a hodnotu dvouiteračního cyklu pro výběr adekvátního výpočtu relativní odchylky koeficientů. Před iterační smyčkou je definice funkce měkkého prahování nazvaná `soft_tresh` podle vztahu (8.6). Maximální počet iterací je omezen parametrem `iteration_max`. Samotný algoritmus se skládá ze čtyř hlavních kroků, které byly popsány v předchozí kapitole. V prvním kroku, kde probíhá měkké prahování, vystupuje parametr `tau` případně vektor váh `w`. Na konci skriptu jsou výpočty jako např. argumenty podmínky, která ukončí algoritmus, pokud se rekonstruované koeficienty příliš neliší od těch, co byly na vstupu.

Varianta, která segmentuje vstupní signál, používá při volání Condatova algoritmu funkci `condat_segment.m`. Ta se liší v tom, že pro zajištění projekce je použita upravená funkce, která násobí oknem i hodnotu `theta`.

9.5.1 Projekce na množinu

Projekce na množinu je realizována funkcí `projection.m` a zajišťuje, aby se dosaďadní výsledek rekonstrukce posunul na hodnotu určenou parametrem `theta`, pokud je na pozicích `Mh` a zároveň je nižší než hodnota `theta`. To samé platí pro vzorky na pozicích `Ml`. Taktéž je zajištěno, aby projekce byla omezena v intervalu $\langle -1, +1 \rangle$. Prakticky je projekce realizovaná pomocí funkcí `max` a `min`.

V případě projekce na množinu u segmentované varianty programu volá funkce `projection_segment.m`. Zde bylo třeba aplikovat řešení, které bude brát v potaz to, že nyní je prahová úroveň `theta` vyjádřena jako vektor násobený analyzujícím oknem.

10 Výsledky testování a měření

Výsledkem praktické části diplomové práce je několik metod pro declipping zvukového signálu, které byly otestovány a pomocí experimentů s úpravou jednotlivých parametrů bylo dosaženo ideálních výsledků pro dané nastavení. Jako objektivní ukazatele kvality rekonstrukce byly použity Δ SDR a PEMO-Q, jako subjektivní ukazatel byl proveden poslechový test MUSHRA.

Nejdříve jsou popsány obecné poznatky, které se týkají testování. Například vzorky hudebních signálů, na kterých bylo testování provedeno a možnosti kontroly během rekonstrukce. Dále zmínka o vizualizaci dosažených výsledků, které slouží spíše jako nástroje pro odhalování nedostatků závažnějšího charakteru.

Následně probíhá vyhodnocení rekonstrukce pomocí ukazatele Δ SDR a posléze i pomocí PEMO-Q včetně diskuze dosažených výsledků. Na objektivní testování navazuje subjektivní testování v podobě poslechového testu MUSHRA a poté probíhá srovnání výsledků poslechového testu a výsledků testu PEMO-Q. V další části jsou shrnuty výsledky testování jednotlivých parametrů algoritmu a na konci je stručné shrnutí všech dosažených výsledků.

10.1 Obecné poznatky k testování

Jelikož jednotlivé parametry algoritmů nemají závislost jen vzhledem k prahové úrovni, ze které dochází k rekonstrukci, ale i vzájemně mezi sebou, bylo hledání optimálního nastavení parametrů založeno spíše na empirickém zkoumání. I z toho důvodu bylo potřeba provést velké množství testů, které pomohly odhalit ideální nastavení pro danou kombinaci.

Příloha obsahuje dokument s tabulkami, které obsahují nejlepší dosažené výsledky pro každou metodu, každý testovaný zvukový signál i pro každý práh θ_c . Tabulka kromě změřených hodnot Δ SDR, počtu iterací a výsledků PEMO-Q obsahuje také konkrétní nastavení parametrů, se kterým byl tento výsledek dosažen. Pro rychlou orientaci mezi kvalitou rekonstrukce jsou pro každý metodu i zvuk výsledné hodnoty všech prahových úrovní θ_c zprůměrovány.

V této kapitole bude nadále používáno označení „Výchozí“ jako označení pro metodu využívající Condatův algoritmus bez segmentace a to z důvodu, že všechny ostatní metody odtud vycházejí a mají shodnou hlavní část programu. Jako zkratka pro psychoakustický model je užíváno „PA model“ a tzv. „2-slope“ je označení pro psychoakustický model, jehož maskovací křivka pro čistý tón má podobu dvou přímk, které připomínají trojúhelník bez základny.

Pro testování byl použit laptop s procesorem Intel i7 4700MQ se základním taktem 2,4 GHz a operační pamětí o velikosti 16 GB a taktem 1600 MHz.

10.1.1 Testovací zvukové signály

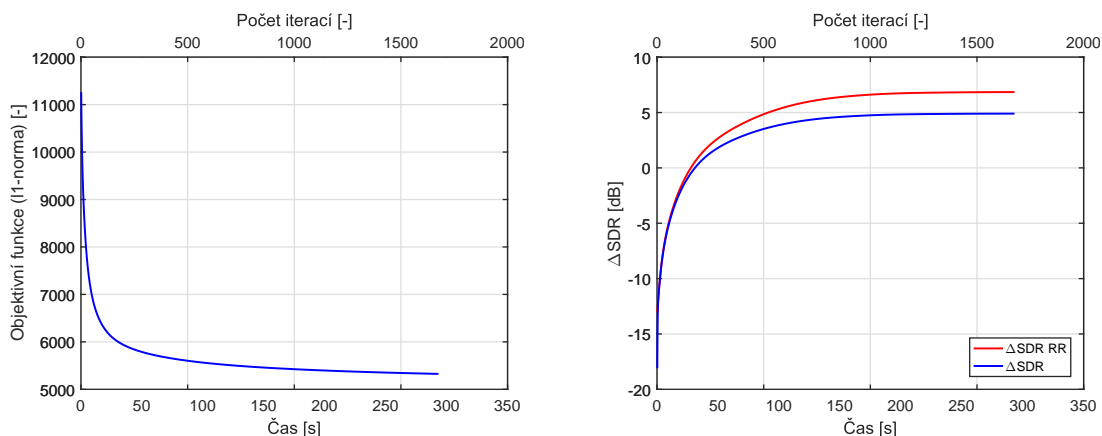
Pro testování bylo použito deset různých zvukových signálů, které obsahují úryvky hry na hudební nástroje. Jedná se o housle, klarinet, fagot, harfu, zvonkohru, čelestu, akordeon, kytaru, klavír a dechový orchestr. Tyto úryvky pocházejí z oficiální databáze EBU [51]. Jednotlivé vzorky byly vybrány tak, aby reprezentovaly škálu obvyklých nástrojů, které se liší svým tonálním charakterem a také svojí řídkostí v kmitočtovém spektru. Všechny tyto vzorky mají vzorkovací kmitočet 44100 Hz, normalizovanou úroveň a jsou uloženy ve formátu WAV. Délka trvání jednotlivých úryvků se pohybuje okolo osmi sekund.

Větší počet odlišných testovacích signálů může poskytnout přesnější závěry o vlastnostech algoritmu. V praxi se ukázalo, že v některých případech pro stejnou metodu a stejné nastavení dochází u různých testovaných signálů k velmi odlišným výsledkům. Před zpracováním rekonstrukčním algoritmem, byla omezena okamžitá hodnota signálu pro prahové úrovně $\theta_c = 0,1, 0,2, \dots, 0,9$.

10.1.2 Průběžné zobrazení výsledků

Průběžné zobrazení výsledků slouží k monitorování chování algoritmu v průběhu rekonstrukce. Lze tak například zjistit, že algoritmus obsahuje chybné nastavení parametrů či to, že nikdy nebude konvergovat k optimálnímu řešení.

Na obrázku 10.1(a) je zobrazen typický průběh objektivní funkce resp. ℓ_1 -normy koeficientů signálu a její minimalizace. Lze si všimnout, že s dalšími iteracemi tato hodnota rychle klesá a postupně se ustaluje na nejnižší hodnotě, které lze minimalizací dosáhnout.



(a) Objektivní funkce v závislosti na čase a počtu iterací.

(b) ΔSDR v závislosti na čase a počtu iterací.

Obr. 10.1: Zobrazení typických průběhů při monitorování chování algoritmu.

Pokud průběh začne během ustalování oscilovat mezi dvěma hodnotami či dojde k prudkému propadu znamená to, že algoritmus nedokáže najít minimum a proto nebude konvergovat k optimálnímu řešení. Doprovodným jevem této situace je poškozený výstupní signál, který obsahuje velké množství nežádoucích zvukových artefaktů.

Na obrázku 10.1(b) je vidět typický průběh ukazatele ΔSDR . Během prvních iterací dochází k prudkému nárůstu ze záporných hodnot a následně k postupnému ustalování na konečné hodnotě. Jakmile je průběh vyrovnaný znamená to, že další iterace nemají význam, protože optimální řešení již bylo nalezeno. Na vodorovné ose dole je stupnice, která odpovídá času rekonstrukce a nahoře počet iterací algoritmu.

Při vyhodnocování je však potřeba sledovat i průběh objektivní funkce, protože existují situace, kdy průběh ΔSDR bude ustálený, zatímco průběh objektivní funkce bude stále oscilovat.

Průběh zobrazený modrou čarou vykresluje ΔSDR celého rekonstruovaného signálu. Průběh grafu v legendě označený jako $\Delta\text{SDR RR}$ pak vykresluje hodnoty ΔSDR po nahrazení vzorků na spolehlivých pozicích. Prakticky se jedná o měření ΔSDR pouze na pozicích signálu, kde docházelo ke clippingu. Vzhledem k tomu, že je při rekonstrukci dovolena odchylka na spolehlivých pozicích, může docházet k poklesu ΔSDR do záporných hodnot, přestože hodnota $\Delta\text{SDR RR}$ bude kladná. Z toho lze usoudit, že čím nižší je ΔSDR vzhledem k $\Delta\text{SDR RR}$, tím více dochází při rekonstrukci k odchylkám na spolehlivých pozicích.

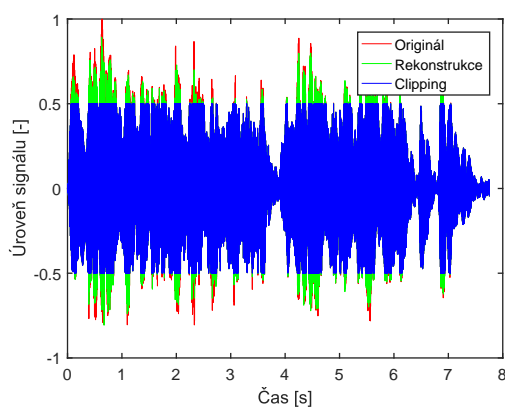
Pokud dochází k těmto odchylkám, rekonstruovaný signál na spolehlivých pozicích neodpovídá vstupnímu signálu a dochází k degradaci. To je důvod, proč jsou tyto vzorky při tvorbě výstupního rekonstruovaného signálu nahrazeny (viz část 9.2.6) a proto jsou i veškeré dále uvedné hodnoty ΔSDR hodnotami až po nahrazení vzorků na spolehlivých pozicích.

10.2 Vizualizace výsledků

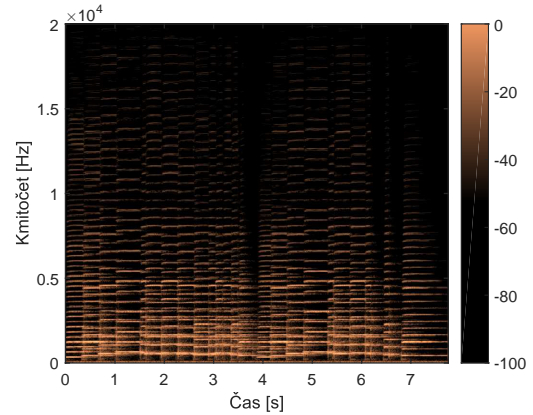
Během testování sloužilo k ověření výsledků i vizuální zobrazení signálů, které navíc umožňuje odhalit některé chyby. Ukazatel ΔSDR například nedokáže odhalit to, zda se v rekonstruovaném signálu nachází nežádoucí kmitočtové složky nebo jestli předpokládaný průběh signálu v časové oblasti vykazuje podobný průběh, jako signál vstupní. To bylo užitečné zejména v počáteční fázi testování při odlaďování jednotlivých algoritmů a odstranování chyb v kódu programu.

Pro zobrazení je tedy vytvořen graf, který obsahuje vstupní, clippovaný i rekonstruovaný signál, tak aby bylo na první pohled zřejmé, zda dochází ke zlepšení či zhoršení. V kmitočtové rovině pro zobrazení slouží spektrogramy (viz část 1.4).

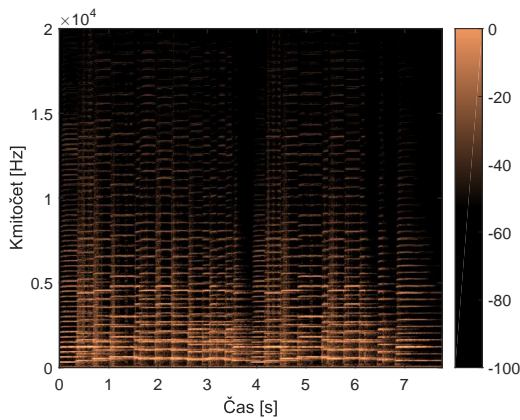
Na obrázcích 10.2(b), 10.2(c) a 10.2(d) je zobrazení spektrogramů originálního, clippovaného a rekonstruovaného signálu. V praxi se ukázalo, že tento způsob zobrazení výsledků testovaných algoritmů je vhodný spíše pro odhalování chyb závažnějšího rázu, které jsou dobře vidět, protože jednotlivé změny v zobrazení jsou poměrně jemné. V tomto případě si lze všimnout, že v čase kolem 4,2 sekundy proběhla oprava resp. odstranění složek, které nebyly součástí originálního signálu. Na druhou stranu v čase přibližně 1,2 sekundy kmitočtové složky vzniklé clippingem nebyly odstraněny, ale lehce zesíleny.



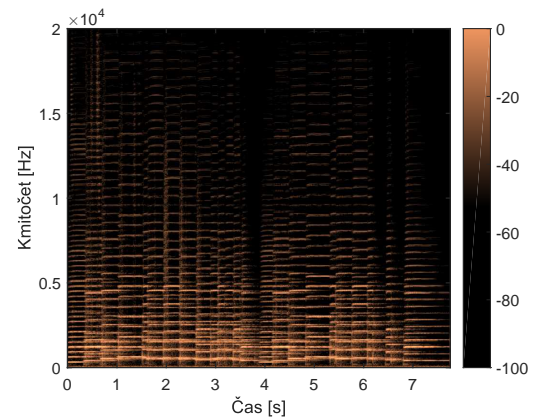
(a) Časový průběh jednotlivých signálů.



(b) Spektrogram originálního signálu.



(c) Spektrogram clippovaného signálu.



(d) Spektrogram rekonstruovaného signálu.

Obr. 10.2: Použité nástroje pro vizualizaci výsledků.

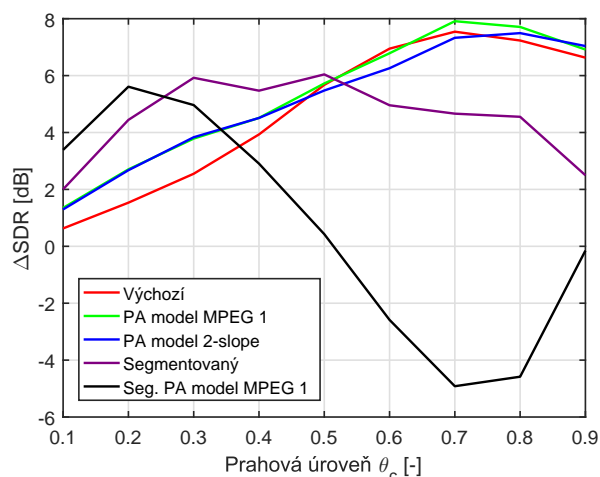
10.3 Vyhodnocení rekonstrukce na základě ΔSDR

Jak již bylo zmíněno v části 4.1.1, tento typ vyhodnocování je frekventovaně používaný. Výpočet je ve srovnání s metodou PEMO-Q triviální a probíhá podle vztahů (4.2) a (4.1). Dává informaci o zlepšení v časové oblasti mezi rekonstruovaným a clippovaným signálem. Je dobré připomenout, že všechny zde uvedené hodnoty ΔSDR

jsou v tomto případě hodnoty až po nahrazení těch vzorků rekonstruovaného signálu, které byly na spolehlivých pozicích. Při testování byla hodnota ΔSDR okamžitě k dispozici a umožňovala rychle odhadnout kvalitu rekonstrukce. Další informace jako poslech či zobrazení na spektrogramu byly dostupné hůře, zvláště při testování většího množství parametrů.

V grafu 10.3 je srovnání jednotlivých metod v závislosti na úrovni θ_c . Průběhy jednotlivých algoritmů obsahují zprůměrované hodnoty ΔSDR všech testovaných ukázek pro daný práh θ_c . Uvedené hodnoty ΔSDR jsou nejlepším dosaženým výsledkem při hledání ideální kombinace parametrů. Ovšem i s ohledem na orientční poslechový test a rychlost konvergence.

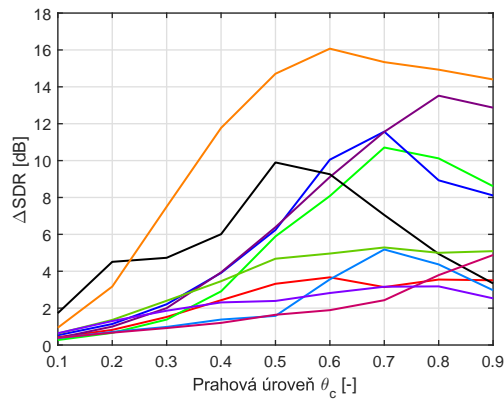
Pro nízké úrovně θ_c dosahují lepšího výsledku metody, které zpracovávají signál po segmentech a naopak pro vysoké úrovně θ_c dosahuje nejlepších výsledků metoda s psychoakustickým modelem MPEG 1. Je možno si všimnout, že rozdíl mezi MPEG 1 a 2-slope se projevuje až při vyšších hodnotách a do $\theta_c = 0,4$ je průběh prakticky totožný. Varianta se segmentací na nízkém prahu θ_c poskytuje nejlepší výsledky, ale s rostoucím θ_c začíná klesat až do záporných hodnot, což svědčí o tom, že nedochází ke zlepšení kvality rekonstruovaného signálu oproti clippovanému.



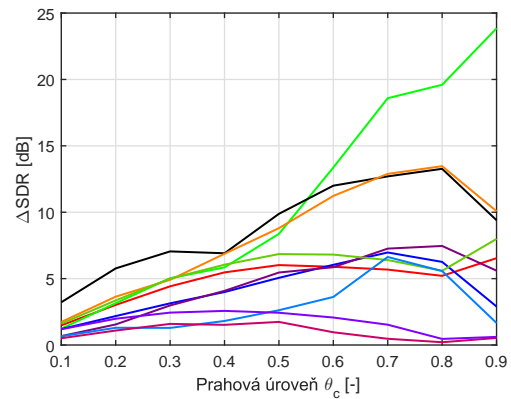
Obr. 10.3: Závislost ΔSDR na úrovni θ_c .

Zobrazení 10.4 obsahuje pět grafů různých metod, které odpovídají závislostem ΔSDR na úrovni θ_c pro jednotlivé testované signály. Nejlepší výsledky pro vyšší θ_c , tedy mírné zkreslení signálu, vykazují algoritmy bez segmentace s psychoakustickým modelem, kde rekonstrukce klarinetu dosahuje hodnot přes 20 dB. Navíc pro tyto algoritmy je mezi jednotlivými zvuky vidět jistá konzistence v průběhu. Při rekonstrukci výchozí variantou Condatova algoritmu překvapuje dosažený výsledek pro nahrávku čelesty. Varianta algoritmu, kde dochází k segmentaci a je použitý PA

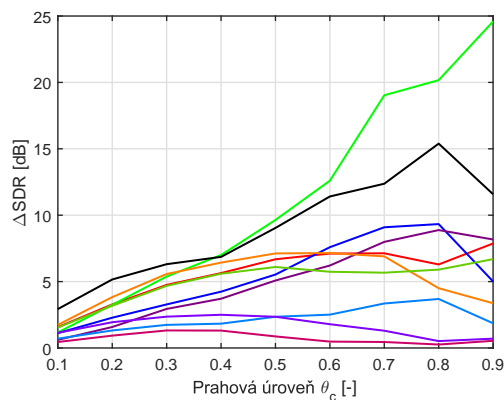
model dává ve srovnání s ostatními algoritmy nejslabší výsledky a jediný rekonstruovaný vzorek, který vykazoval pro všechny prahy θ_c kladnou hodnotu ΔSDR je nahrávka dechového orchestru. Patrně nejméně konzistentní je segmentovaný Condatův algoritmus, protože výsledky pro jednotlivé zvukové nahrávky se velmi liší a v případě kytary se výsledek pro $\theta_c = 0,8$ pohybuje v záporných hodnotách.



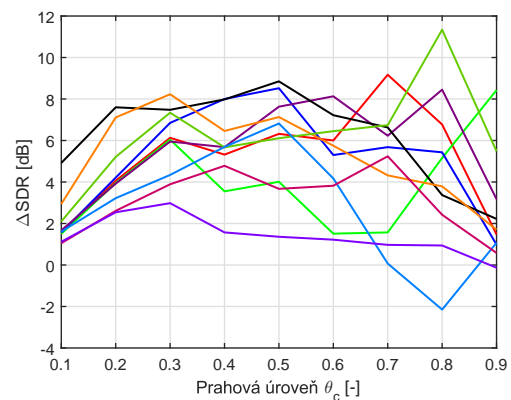
(a) Výchozí metoda.



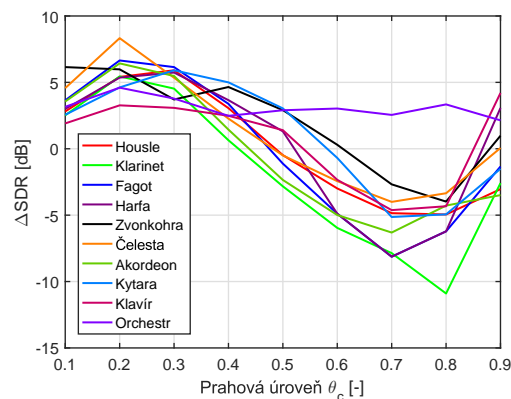
(b) PA model MPEG 1.



(c) PA model 2-slope.



(d) Metoda se segmentací.



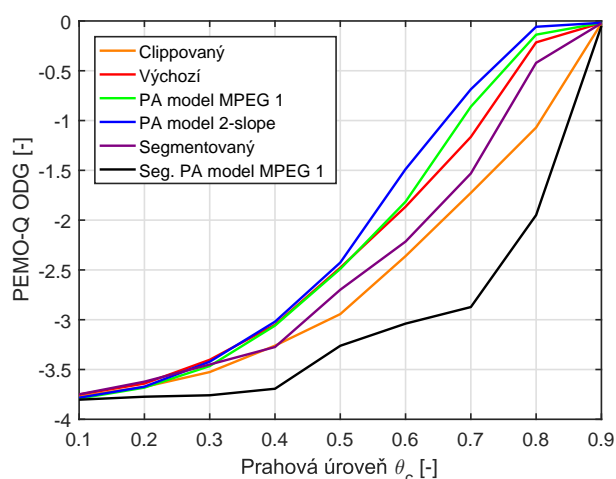
(e) Metoda se segmentací a PA modelem MPEG 1.

Obr. 10.4: Závislost ΔSDR na úrovni θ_c pro jednotlivé metody.

10.4 Vyhodnocení rekonstrukce na základě PEMO-Q

V této části budou výsledky rekonstrukce vyhodnoceny pomocí metody PEMO-Q, která hodnotí kvalitu rekonstrukce pomocí modelu slyšení tak, aby výsledek hodnocení koreloval s hodnocením kvality při poslechu (viz část 4.1.2). Struktura části je shodná s vyhodnocením pomocí ukazatele ΔSDR , tzn. nejdříve budou porovnány jednotlivé algoritmy a poté i jednotlivé hudební ukázky v rámci každého z testovaných algoritmů. PEMO-Q navíc umožňuje i naměřit hodnoty pro clippovaný signál a z grafů tak lze vyčíst také zlepšení vzhledem k signálu před rekonstrukcí. Z porovnání výsledků ΔSDR a PEMO-Q vyplývá, že ukazatel ΔSDR nutně nekorresponduje s PEMO-Q, které simuluje vlastnosti lidského slyšení.

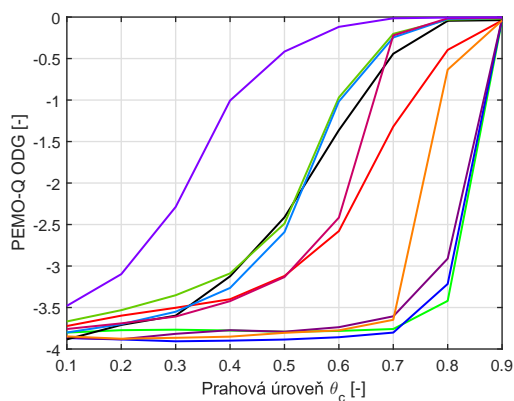
Na obrázku 10.5 jsou vidět jednotlivé průběhy testovaných algoritmů v závislosti na prahové úrovni θ_c . Modře zobrazený průběh, tedy Condatův algoritmus s PA modelem 2-slope vykazuje celkově nejlepší výsledky, ovšem do úrovně $\theta_c = 0,5$ je průběh takřka shodný s modelem MPEG 1 i s algoritmem bez PA modelu. Nejhůře dopadla segmentovaná varianta algoritmu s PA modelem MPEG 1. Při pozorování průběhů lze vidět, že od hodnoty $\theta_c = 0,3$ mají rekonstrukce až na jeden případ vyšší hodnocení, než clippovaný signál, a tudíž dochází ke zlepšení kvality při rekonstrukci.



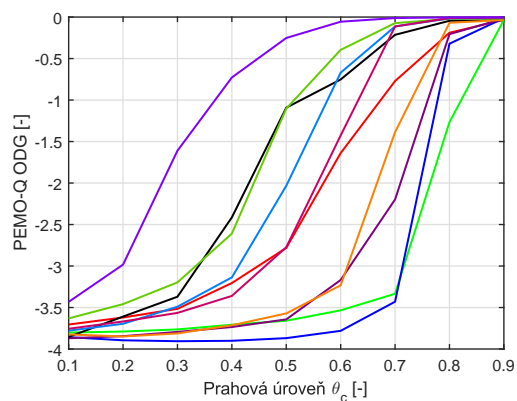
Obr. 10.5: Závislost hodnoty na stupnici ODG na úrovni θ_c .

Zobrazení 10.6 obsahuje šest různých grafů, kde jsou vyhodnoceny výsledky testu PEMO-Q na stupnici ODG v závislosti na hodnotě θ_c . Kromě pěti testovaných metod je zde i graf obsahující průběhy pro clippované signály pro lepší komparaci výsledků. Ačkoliv na první pohled by se mohlo zdát, že nejvyšší kvality při rekonstrukci dosahuje nahrávka dechového orchestru, z pohledu na průběh clippovaného signálu dechového orchestru vyplývá, že zlepšení je jen nepatrné. Největší zlepšení u výchozí metody podává signál kytary a čelesty, naopak nejhůře opraven byl signál fagotu a klarinetu, u kterých zlepšení nastává až pro mírné zkreslení nad $\theta_c = 0,6$.

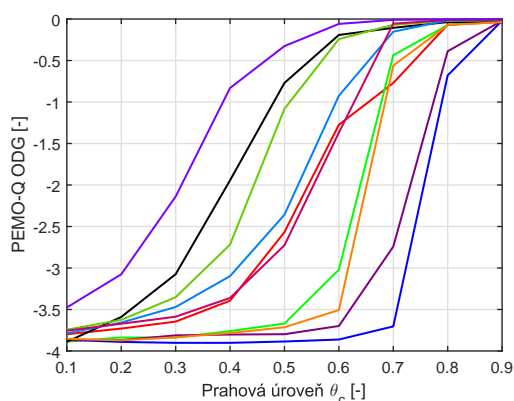
Oba modely psychoakustického modelu podávají téměř shodné výsledky a ke zlepšení dochází ve všech případech jak vzhledem ke clippovanému signálu tak vzhledem k rekonstrukci bez PA modelu. Pro segmentovanou variantu dochází k mírnému zlepšení, jen v případě signálu zvonkohry a akordeonu a pro segmentovanou metodu s PA modelem dochází ve všech případech naopak ke zhoršení kvality signálu.



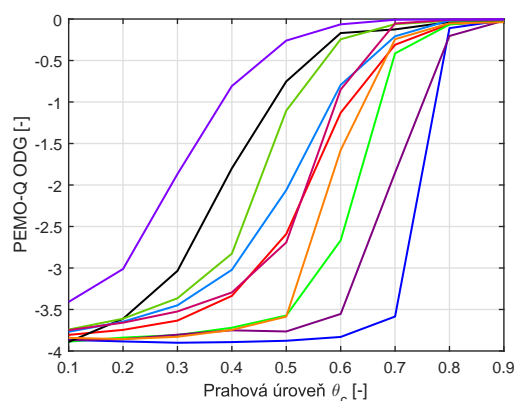
(a) Clippovaný signál.



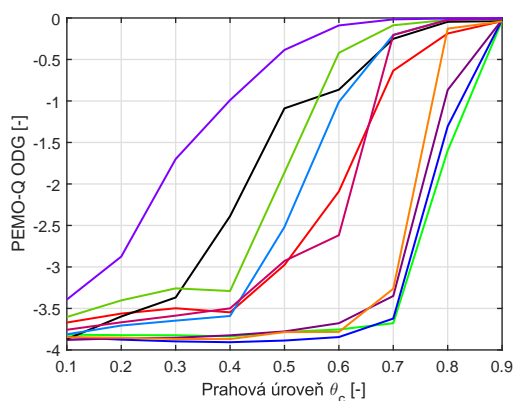
(b) Výchozí metoda.



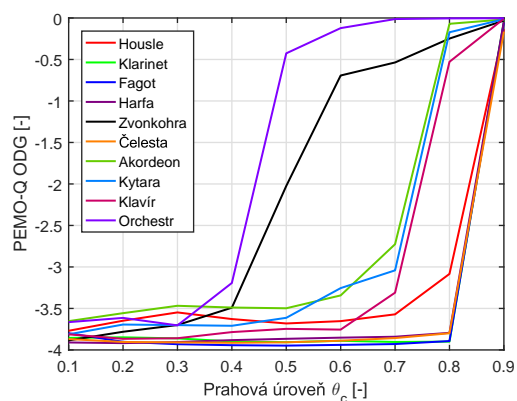
(c) PA model MPEG 1.



(d) PA model 2-slope.



(e) Metoda se segmentací.



(f) Metoda se segmentací a PA modelem MPEG 1.

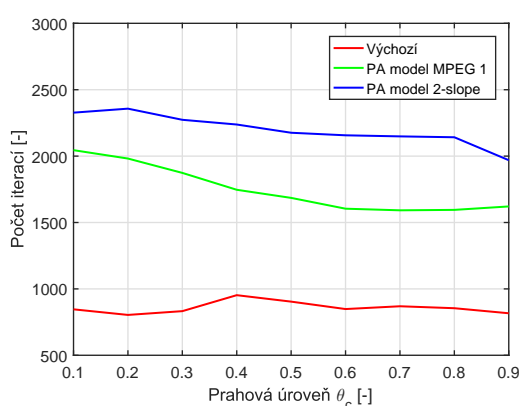
Obr. 10.6: Závislost ODG na úrovni θ_c pro jednotlivé metody.

10.5 Srovnání doby rekonstrukce

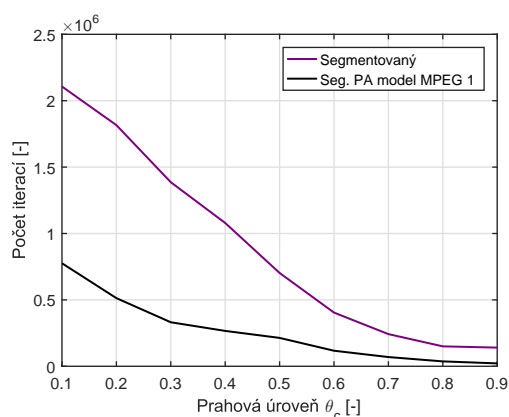
Každý z algoritmů nacházel řešení s různou rychlostí. Při testování byl brán ohled na nastavení maximálního počtu iterací tak, aby při výpočtu docházelo k ukončení algoritmu spíše na základě překročení terminačního kritéria, které tak provede při relativní odchylce koeficientů 0,00001 mezi iteracemi, než při dosažení maximálního počtu iterací.

Pro srovnání algoritmů na základě rychlosti zde bylo využito znalosti počtu iterací, které byly potřebné pro konvergenci, resp. ke splnění výše uvedených kritérií. Vzhledem k odlišnému přístupu probíhá zvlášť porovnání algoritmů, které zpracovávají signál po segmentech a které jej zpracovávají najednou. Při hledání optimálních výsledků neměl počet iterací takovou prioritu, ovšem v některých případech docházelo k tomu, že bylo k dispozici řešení s podobnou hodnotou ΔSDR ale rozdíl v potřebném počtu iterací se lišil v řádu stovek.

Na obrázku 10.7(a) je rozdíl pro metody, které při rekonstrukci nemají segmentovaný signál. V rychlosti dominuje výchozí metoda a je možno si všimnout, že po zprůměrování počtu iterací všech testovaných zvuků pro různé nastavení θ_c zde není žádná zjevná závislost. Pro metody s psychoakustickým modelem platí, že počet potřebných iterací ke konvergenci s rostoucí θ_c mírně klesá. Nejhuře zde dopadl PA model 2-slope. Je důležité zmínit, že reálná rychlost konvergence metod s PA modelem je ještě menší, protože do výpočtu není zahrnutá doba, během které docházelo k prvotní rekonstrukci bez GMT. Z pohledu na obrázek 10.7(b) je patrná silná závislost na aktuální prahové hodnotě θ_c a výrazný rozdíl v rychlosti výpočtu mezi metodami. Obecně i přes odlišný způsob zpracování je možno říci, že nejrychlejší je výchozí metoda a naopak nejpomalejší je varianta se segmentací.



(a) Metody bez segmentace.



(b) Metody se segmentací.

Obr. 10.7: Závislost počtu iterací na úrovni θ_c pro jednotlivé metody.

10.6 Poslechový test MUSHRA

Nedílnou součástí vyhodnocení kvality rekonstrukce zvukových signálů je i subjektivní hodnocení pomocí poslechových testů. Narozdíl od vyhodnocení pomocí objektivních ukazatelů, v tomto případě nebudou testovány všechny úrovně clippingu θ_c , ale pouze $\theta_c = 0,1$ a $\theta_c = 0,5$.

10.6.1 Metoda testu

Jako metoda poslechového testu byl zvolena MUSHRA (viz část 4.2.1). Ačkoliv byl tento typ testu původně vyvinutý pro subjektivní testování kvality ztrátových kodeků, hodí se i pro potřeby zjišťování kvality rekonstrukce clippovaných signálů. Test kromě možnosti zkoumat více rekonstruovaných signálů najednou obsahuje také skrytý referenční signál, který odpovídá vstupnímu signálu před clippingem. Dále test ještě obsahuje tzv. kotvu, která je v tomto případě clippovaný signál, tedy nejhorší výsledek, ze kterého se vychází.

10.6.2 Aplikace Scale

Jako prostředí pro testování byla zvolena aplikace Scale, která je spuštěna v Matlabu a má své vlastní jednoduché grafické rozhraní. Autoři pocházejí z německé univerzity TH Köln. Aplikace je volně ke stažení na jejich webových stránkách <http://audiogroup.web.th-koeln.de/scale.html>. Scale nabízí prostředí pro vytvoření testů, kde lze nastavit tzv. scénáře testů. Vzhledem k tomu, že testováno bylo 10 zvukových signálů, přitom každý na 2 úrovních θ_c , test obsahuje 20 různých scénářů. Každý scénář obsahuje jeden zvuk, který byl rekonstruován pomocí pěti různých algoritmů, čistý vstupní signál jako referenci a clippovaný signál jako kotvu. Celkem tedy 7 nahrávek v jednom scénáři.

Aby byla zajištěna správnost vyhodnocení testu, je potřeba při vytváření jednotlivých scénářů vkládat testované zvuky ve stejném pořadí, tj. například aby reference byla pokaždé do scénáře zadána jako první. V tomto pořadí jsou pak změřená data ukládána do vektorů pro jednotlivé scénáře.

Aplikace umožňuje také analýzu naměřených dat. Dokáže zobrazit data libovolného počtu respondentů, či scénářů testování. Užitečné je zobrazení do grafů se zobrazením jednotlivých průběhů, jejich střední hodnotou či kombinací obou možností. Výhodou této aplikace je formát výsledků, který je uložen v Matlab souboru s příponou .mat, a lze jej pohodlně využít pro další práci s naměřenými daty.

10.6.3 Podmínky a průběh testu

Testu se zúčastnilo celkem 20 respondentů. Při výběru respondentů byl brán ohled na výběr různorodého spektra lidí. Testování proběhlo s laptopem, jehož parametry jsou uvedeny na začátku této kapitoly. Jako zvuková karta byla použita Lexicon Alpha a pro poslech posloužila uzavřená sluchátka Sennheiser HD6 mix. Poslechový test probíhal v místnosti s minimalizovanými ruchy z okolí a s konstantním nastavením výstupní úrovně na zvukové kartě.

Před samotným testem byl každému účastníkovi vysvětlen záměr této práce a zdůvodnění potřeby subjektivního testu. Dále byli v rámci potřeb seznámeni s metodou MUSHRA, s prostředím, ve kterém byl spuštěn test a bylo jim sděleno, co se od nich vlastně v testu očekává. Mimoto byl na ukázkovém případě respondentům předveden příklad rozdílu mezi referencí a kotvou, aby získali větší představu o tom, v jakých mezích se budou rekonstruované signály pohybovat.

Respondent měl na stupnici od 0 do 100 určit podobnost testovaného signálu s referenčním. Referenční signál bylo možné přehrát kdykoliv během testování. Testované signály byly ukryty pod jednotlivými písmeny A až G a hodnocení probíhalo pomocí posuvníku, který umožňuje intuitivnější zadání hodnoty oproti výběru na pětibodové stupnici. Průměrný čas testu se pohybuje mezi 25 a 30 minutami.

Aby bylo možné porovnávat získaná data s měřením pomocí ukazatele PEMO-Q, bylo třeba je upravit do stejné podoby tj. na stupnici ODG (viz část 4.1.2). Jednotlivé hodnoty poslechového testu byly vyděleny 25 a z výsledku bylo odečteno 4. Nyní se stupnice pohybuje v rozsahu -4 až 0.

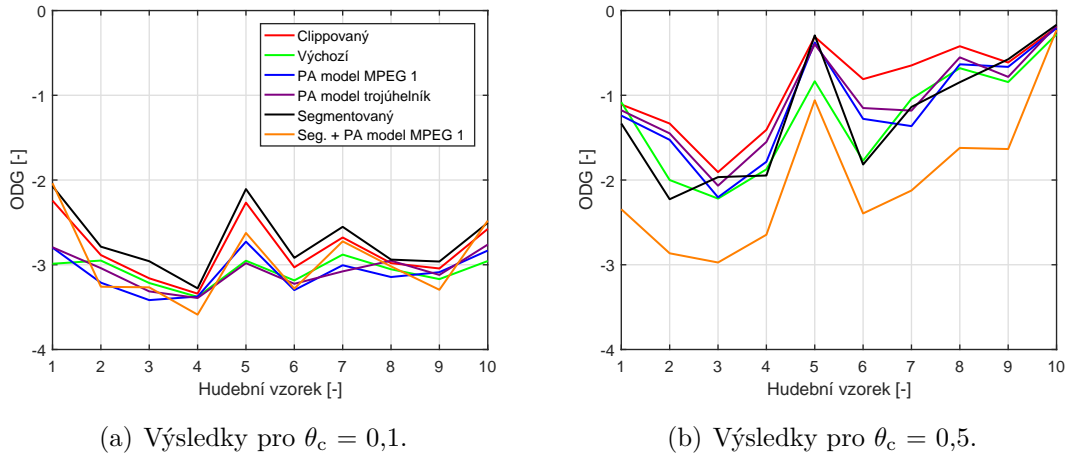
10.6.4 Výsledky testu

Naměřená data všech respondentů byla zprůměrována pro každý scénář a vynesena do grafů jako závislost hodnocení na stupnici ODG na testovaném hudebním vzorku. Data pro $\theta_c = 0,1$ a $\theta_c = 0,5$ byla zobrazena v samostatných grafech. Změřené hodnoty každého z respondentů jsou uloženy v tabulkách jako příloha v podobě souboru s příponou .xlsx. Soubor na každém z 20 listů pro každý úryvek i práh θ_c obsahuje data všech subjektů pro všechny testované zvuky. Takto lze například zjistit, že v několika případech nedošlo k rozpoznání rozdílu mezi rekonstrukcí a referenčním signálem.

Na obrázku 10.8(a) jsou k vidění jednotlivé průběhy pro každý testovaný algoritmus a také pro clippovaný signál. Na první pohled je zřejmé, že signály na prahu $\theta_c = 0,1$ prakticky nejsou opraveny a kvalita je hodnocena jako špatná. V tomto případě jako nejpříjemnější vychází segmentovaný Condatův algoritmus, zatímco algoritmy s psychoakustickým modelem způsobují při rekonstrukci velmi silného zkreslení vznik nežádoucího zvukového artefaktu, který vzdáleně připomíná hudební

efekt phaser. Tento jev doprovázel všechny rekonstrukce s PA modelem pro $\theta_c = 0,1$ bez zásadního vlivu nastavení parametrů algoritmu. Nejlépe hodnocená ukázka byla zvonkohra a nejhůře harfa.

Obrázek 10.8(b) zobrazuje jednotlivé průběhy při nastavení $\theta_c = 0,5$, kde je vidět zlepšení, přesto signál poškozený clippingem v mnoha případech vychází lépe, než rekonstrukce. Tyto průběhy ukazují na důležitost subjektivního testování, které může odhalit nedostatky nehledě na přijatelné výsledky pomocí ukazatele ΔSDR .



Obr. 10.8: Výsledek testu MUSHRA pro jednotlivé metody.

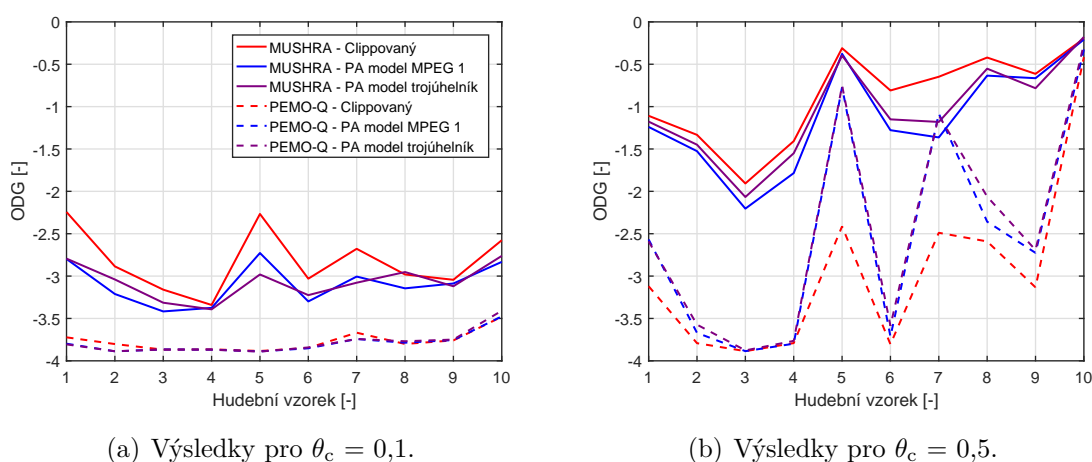
V tabulce 10.1 se nachází vysvětlivka číselného označení testovaných signálů, které bylo použito v grafech 10.8(a) a 10.8(b).

Tab. 10.1: Číselné označení použitých testovacích hudebních ukázek.

Označení	Název zvukové ukázky
1	Housle
2	Klarinet
3	Fagot
4	Harfa
5	Zvonkohra
6	Čelesta
7	Akordeon
8	Kytara
9	Klavír
10	Dechový orchestr

10.7 Porovnání výsledků testu MUSHRA a PEMO-Q

V této části dochází k porovnání dat získaných subjektivním poslechovým testem a objektivním testováním pomocí ukazatele PEMO-Q. Při pohledu na následující obrázky je patrná jistá korelace mezi výsledky. Pro $\theta_c = 0,1$ je zásadnější rozdíl snad jen při hodnocení zvonkohry. Obecně je výsledek PEMO-Q horší, než data získaná subjektivním testováním. Pro $\theta_c = 0,5$ byl zjištěn překvapivý rozdíl mezi hodnocením pomocí MUSHRA a PEMO-Q v případě akordeonu a zvonkohry. Dle PEMO-Q rekonstrukce vykazuje výrazné zlepšení, zatímco dle poslechového testu v případě zvonkohry není téměř rozdíl a pro akordeon dokonce vykazuje zhoršení oproti clipovanému.



Obr. 10.9: Porovnání výsledku PEMO-Q a testu MUSHRA pro jednotlivé metody.

10.8 Testování parametrů algoritmu

V této části jsou závěry ohledně nastavení různých parametrů algoritmu. Je poukázáno na rozdíly v nastavení, pokud algoritmus obsahuje i psychoakustický model.

Jako ukazatel zde posloužilo ΔSDR a orientační poslechový test vygenerovaného zvukového souboru. Dalším ukazatelem byl průběh objektivní funkce a ΔSDR v jednotlivých iteracích algoritmu.

Testy pro parametry γ , σ a ρ byly provedeny jen na výchozím Condatovu algoritmu bez segmentace a pro Condatův algoritmus s psychoakustickým modelem MPEG 1 a 2-slope. V případě porovnání segmentované a nesegmentované varianty algoritmu už není změna v nastavení parametrů tak zásadní, jako v případě rozdílu při použití či nepoužití psychoakustického modelu pro rekonstrukci signálu.

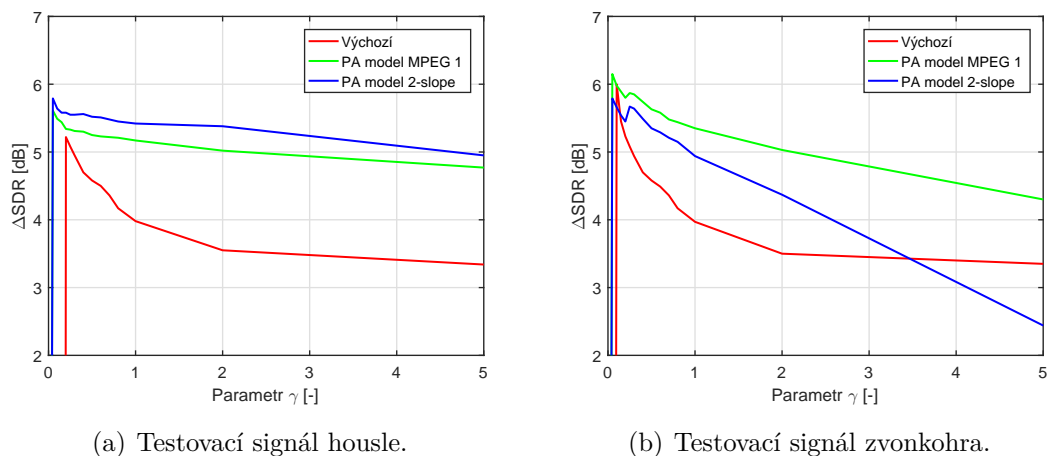
Pokud je použita nevhodná kombinace hodnot parametrů, začne se během prvních několika desítek iterací aktuální hodnota ΔSDR hluboko propadat do zápor-

ných hodnot a dojde k předčasnému ukončení rekonstrukce. Tím se při testování ušetří čas, protože v takovém případě algoritmus nebude konvergovat.

Dva grafy pro zobrazení závislosti jednoho parametru jsou použity z toho důvodu, že přesný průběh se pro různé testované signály liší. Existuje mezi nimi ale podobnost, na základě které lze přibližně odhadnout vliv tohoto parametru na velikost ΔSDR . Na základě tohoto principu jsou zobrazeny i další grafy pro ostatní parametry po dvojicích. Pro reprezentaci byly vybrány dva spektrálně odlišné zvuky, a to housle a zvonkohra.

10.8.1 Parametr γ

Pomocí parametru γ lze nastavit míru, jak velký bude mít vliv povolená odchylka na spolehlivých pozicích na hledané řešení. Prakticky bylo zjištěno, že tento parametr má největší vliv na ukazatel ΔSDR . Nejčastěji se optimální hodnoty nacházely v rozsahu 0,05 až 0,2 v závislosti na metodě a testovaném signálu.



Obr. 10.10: Závislost ΔSDR na velikosti parametru γ .

Na obrázcích 10.10(a) a 10.10(b) je vykreslena závislost ΔSDR na velikosti parametru γ pro tři odlišné metody. Na první pohled je zřejmý rozdíl v chování parametru γ pro metodu bez a s psychoakustickým modelem. Směrem od vysokých hodnot γ k nízkým dochází k nárůstu ΔSDR . Pro metody s psychoakustickým modelem je tento nárůst poměrně lineární. Po dosažení kritické hodnoty parametru γ metoda přestává fungovat a dochází k prudkému propadu. Paradoxně platí, že v bodě, kde γ poskytuje nejlepší ΔSDR , neleží optimální výsledek. Při takovémto nastavení dochází k rozkmitání průběhu objektivní funkce a z toho vyplývá, že nikdy nebude konvergovat. Stejně tak dochází k nárůstu slyšitelnosti negativních zvukových artefaktů. Optimální hodnota parametru γ je tedy o něco vyšší, než ta, při které je

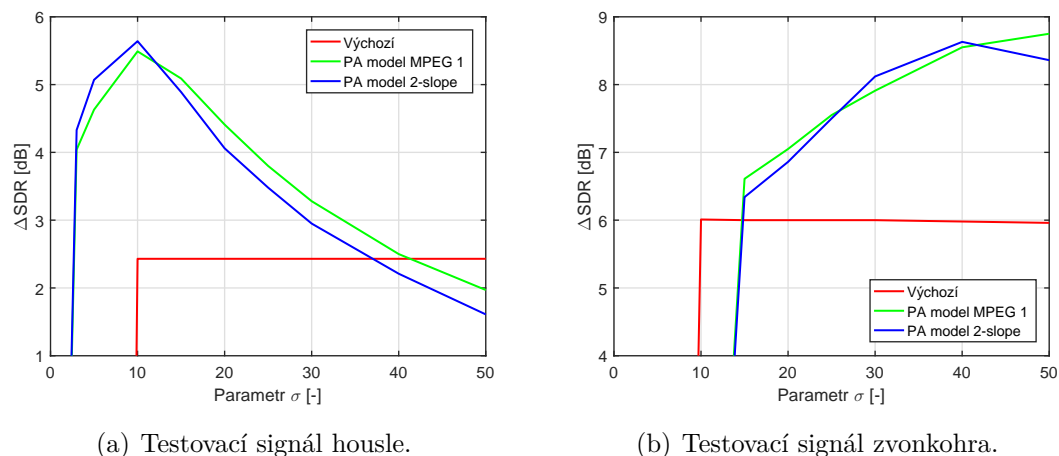
nejlepší ΔSDR . Přesnou hodnotu je třeba empiricky nastavit. Bylo také zjištěno, že s rostoucím parametrem γ obvykle narůstá doba potřebná ke konvergenci algoritmu.

Z obrázků je také patrné, že psychoakustický model umožňuje nastavit menší hodnotu γ . To se odvíjí od toho, že pro konvergenci algoritmu s PA modelem je nutné nastavit nižší ρ , které do jisté míry umožní použít nižší γ a přitom stále dojít k optimálnímu řešení.

10.8.2 Parametr σ

Cílem zobrazení závislosti ΔSDR na nastavení parametru σ je důkaz, že při implementaci PA modelu do Condatova algoritmu začíná mít tento parametr vliv na kvalitu rekonstrukce. Optimální nastavení parametru σ je velmi závislé na konkrétním zvukovém signálu a částečně i na parametru θ_c . S rostoucí θ_c klesá počet iterací, které jsou potřebné ke splnění terminačního kritéria.

Pro výchozí metodu platí, že změna σ nemá přímý vliv na získanou hodnotu SDR ani na rychlost konvergence. Na druhou stranu vyšší σ umožňuje použít vyšší hodnotu parametru ρ , které má příznivý vliv na rychlost rekonstrukce.

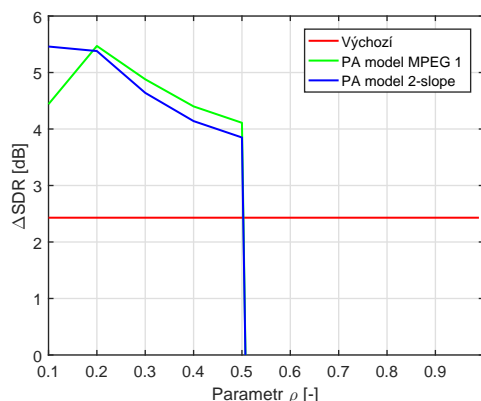


Obr. 10.11: Závislost ΔSDR na velikosti parametru σ .

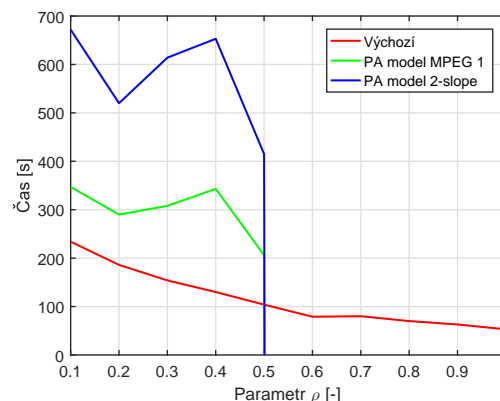
10.8.3 Parametr ρ

Parametr ρ v rámci algoritmu ovlivňuje velikost kroku, s jakým dochází k hledání řešení problému. Při pohledu na obrázek 10.12(a) je opět vidět značný rozdíl v chování metod. Parametr ρ u výchozí metody bez PA modelu nemá vliv na hodnotu ΔSDR , pouze se s jeho rostoucí hodnotou snižuje čas potřebný ke konvergenci (viz obrázek 10.12(b)). Nebyl vypořádan ani vliv při poslechu rekonstruovaných nahrávek. Proto je vhodné volit tuto hodnotu co nejvyšší. Při použití PA modelu nelze

použít $\rho > 0,5$ a optimální hodnota z hlediska rychlosti i ukazatele ΔSDR činí 0,2. Tato skutečnost přispívá k nárůstu času potřebného pro rekonstrukci.



(a) Závislost ΔSDR na velikosti ρ .



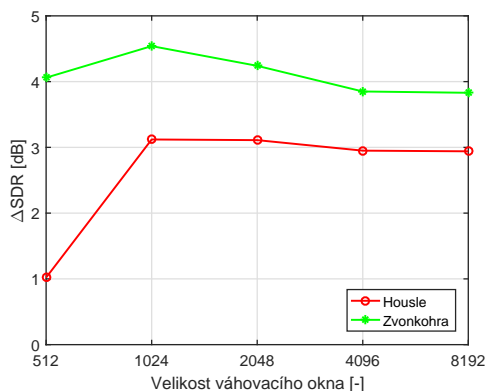
(b) Závislost doby rekonstrukce na ρ .

Obr. 10.12: Závislost ΔSDR a doby rekonstrukce na velikosti parametru ρ .

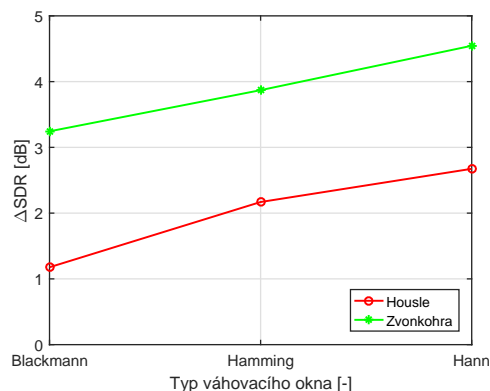
10.8.4 Váhovací okno

Velikost váhovacího okna byla nastavena na 1024 vzorků. Při použití menšího okna již dochází k zdatelnému propadu hodnoty ΔSDR , jak lze vidět na obrázku 10.13(a). Větší okno sice dává podobný výsledek, ale není tolik žádoucí vzhledem k tomu, že maskovací křivka PA modelu je tím přesnější, čím je okno menší. Faktem zůstává, že s rostoucí velikostí okna klesá čas potřebný ke konvergenci algoritmu.

Závislost ΔSDR na typu váhovacího okna na obrázku 10.13(b) ukazuje, že nejlepší možností je použití Hannova váhovacího okna. Otestované Blackmannovo a Hammingovo okno ani v jednom případě neposkytlo lepší výsledek, proto je u všech testovaných metod použito právě Hannovo okno.



(a) Závislost ΔSDR na velikosti okna.

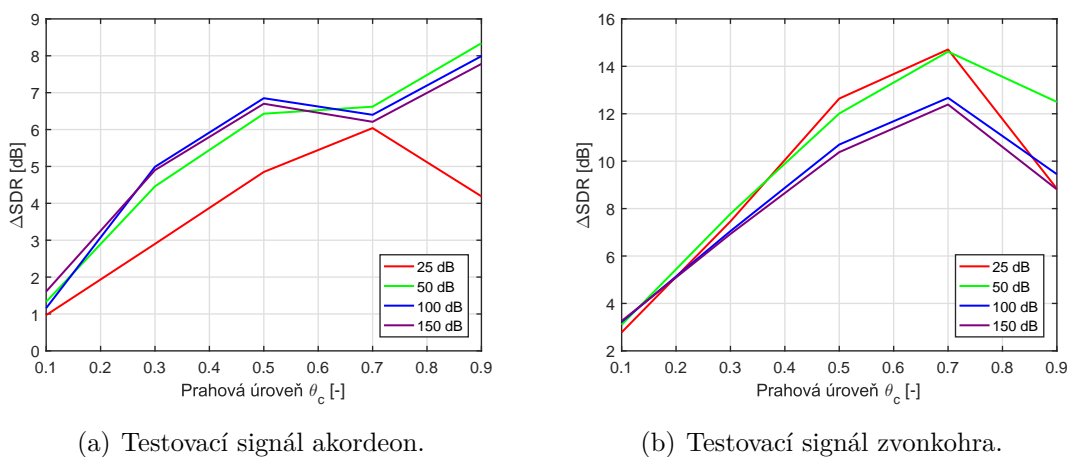


(b) Závislost ΔSDR na typu okna.

Obr. 10.13: Vliv vlastností váhovacího okna na ΔSDR .

10.8.5 Dynamický rozsah

Dynamický rozsah v tomto pojetí představuje rozsah, v jakém se pohybují hodnoty GMT. Z pohledu na obrázky 10.14(a) a 10.14(b) lze usoudit, že pro signály, které jsou více řídké je vhodnější menší dynamický rozsah, zatímco pro signály, které příliš řídké nejsou, je vhodnější větší dynamický rozsah. Jako kompromisní nastavení byla použita hodnota 100 dB.



Obr. 10.14: Závislost ΔSDR na velikosti θ_c pro různé nastavení dynamického rozsahu.

10.9 Shrnutí dosažených výsledků

Z hlediska hodnocení na základě ΔSDR podává nejlepší výsledky metoda s PA modelem MPEG 1, přičemž rozdíl mezi PA modelem 2-slope se týká pouze vyšších hodnot θ_c . Z výsledků testování pomocí PEMO-Q je patrné, že nejlepší rekonstrukce bylo dosaženo metodou s PA modelem 2-slope, a že stejně jako při hodnocení pomocí ΔSDR jsou průběhy pro nízké hodnoty θ_c téměř totožné. V obou případech dochází ke zlepšení oproti použití metody bez psychoakustického modelu za cenu zvýšení doby potřebné pro rekonstrukci. Bohužel, varianta se segmentací a psychoakustickým modelem podává ve většině případů nedostatečné výsledky.

Z orientačních poslechových testů bylo odhaleno, že většina rekonstruovaných nahrávek měla i přes mírně odlišné hodnoty ΔSDR podobný charakter, a že případné nedostatky se obvykle objevovaly na stejných pozicích v signálu, přestože například časové průběhy těchto signálů se v rámci jednotlivých testů mírně lišily.

Na základě poslechového testu MUSHRA bylo zjištěno, že pro silné zkreslení při $\theta_c = 0,1$ je rekonstrukce ve většina případů nedostatečná a clippovaný signál byl často hodnocen jako přijatelnější, než rekonstrukce. Výjimkou byla segmentovaná

varianta bez PA modelu. Pro vyšší práh docházelo k subjektivnímu zlepšení pouze u dvou testovaných hudebních ukázek.

Napříč všemi metodami bylo nejlepších výsledků dosaženo u nahrávky zvonkohry, která má dostatečně malou řídkost signálu natolik, že umožňuje kvalitnější rekonstrukci. Naopak nejslabších výsledků dosahovala rekonstrukce poškozených nahrávek fagotu, klavíru a harfy.

Segmentovaná varianta algoritmu má velmi dlouhou dobu rekonstrukce, v porovnání se svým nesegmentovaným protějškem trvá rekonstrukce přibližně 10 krát déle a nebylo neobyklé, že rekonstrukce jednoho signálu o délce přibližně osm sekund trvala okolo půl hodiny.

11 Závěr

Cílem práce bylo zejména nastudování problematiky znehodnocení zvukového signálu jevem, který se nazývá clipping a možnosti rekonstrukce takto znehodnocených signálů pomocí metody založených na řídkých reprezentacích signálu. V rámci teoretické části byl také nastudován psychoakustický model MPEG a zjištěny možnosti jeho využití pro zvýšení kvality declippingu.

V prostředí Matlab byly implementovány algoritmy pro declipping, které byly rozšířeny o psychoakustický model. Následně proběhlo experimentální hledání optimálních hodnot parametrů za účelem dosažení co nejlepších výsledků. Tyto výsledky byly poté vyhodnoceny pomocí dvou objektivních ukazatelů a subjektivního poslechového testu.

První kapitola se věnuje samotné problematice clippingu. Druhá kapitola popisuje proces declippingu a stěžejní metody k řešení tohoto problému, které byly v minulosti představeny. V kapitole číslo 3 s názvem Psychoakustika jsou popsány psychoakustické jevy, které budou později využity v implementovaném algoritmu. V následující kapitole jsou popsány základní metody hodnocení kvality restaurace signálů, pomocí kterých bude zjišťována objektivní i subjektivní kvalita restaurovaných signálů.

V dalších dvou kapitolách jsou pak rozebrány základní teoretické znalosti z oblasti řídkých reprezentací, na které navazuje kapitola o proximálních algoritmech. V té jsou kromě obecného vysvětlení uvedeny tři algoritmy, které lze použít pro řešení úlohy declippingu. Další kapitola se věnuje matematické formulaci obecné úlohy declippingu a následně konkrétnímu řešení pomocí Condatova algoritmu. Tato kapitola obsahuje i řešení s využitím psychoakustického modelu. Programové řešení úlohy obsahující popis kódu programu je k nalezení v kapitole 9.

V poslední kapitole jsou popsány způsoby, pomocí kterých byly implementované metody založené na Condatovu algoritmu otestovány. Celkem bylo vytvořeno pět metod. První metoda využívá Condatův algoritmus a zpracovává signál najednou, čili bez segmentace. Druhá metoda navíc obsahuje psychoakustický model MPEG 1, třetí pro změnu model 2-slope. Čtvrtá metoda je Condatův algoritmus upravený pro zpracování segmentovaného vstupního signálu a pátá kromě segmentace využívá také psychoakustický model MPEG 1. Rekonstruované signály byly vyhodnoceny pomocí objektivních ukazatelů ΔSDR a PEMO-Q, pro subjektivní vyhodnocení byl proveden poslechový test MUSHRA s 20 respondenty. V grafech jsou vyneseny závislosti průměrných hodnot ΔSDR každé metody na nastavení prahové úrovně θ_c . Dle tohoto ukazatele podává celkově nejlepší výsledky metoda založená na Condatovu algoritmu s psychoakustickým modelem MPEG 1. Z pozorování výsledků měření PEMO-Q je patrné, že nejlepší rekonstrukce bylo dosaženo pomocí metody s psy-

choakustickým modelem 2-slope, a že stejně jako při hodnocení pomocí ΔSDR jsou průběhy pro nízké hodnoty θ_c téměř totožné. Výsledky poslechového testu MUSHRA vykazují ve většině případů podprůměrné hodnoty kvality rekonstrukce. Při porovnání výsledků PEMO-Q a MUSHRA je možné vidět jistou korelaci, přestože v některých bodech se výsledky zásadně liší.

Součástí přílohy je dokument, který obsahuje výsledky pro všechny testované metody a testovací zvukové signály omezené na 9 prahových úrovních θ_c . Tabulky kromě změřených hodnot ΔSDR , počtu iterací potřebných ke konvergenci algoritmu a výsledků PEMO-Q na stupnici ODG obsahují také konkrétní nastavení parametrů, se kterými byl tento výsledek dosažen. Pro rychlou orientaci mezi kvalitou rekonstrukce byly pro každou metodu i zvukový signál výsledné hodnoty všech prahových úrovní zprůměrovány.

Potenciální pokračování této práce by mohlo spočívat v dalších experimentech s psychoakustickým modelem, kdy například v nově vydaném článku [52] bylo zjištěno, že při použití prosté paraboly namísto globálního maskovacího prahu lze dosáhnout zlepšení ΔSDR v řádu jednotek dB. Ke zlepšení došlo také na stupnici ODG v testu PEMO-Q.

Literatura

- [1] ABEL, J.; SMITH, J. *Restoring a clipped signal*. Proc. Int. Conf. Acoustics, Speech and Signal Processing(ICASSP). 3rd ed, 1991. s. 1745–1748. [cit. 2018-11-18]. Dostupné z: <<http://ieeexplore.ieee.org/document/150655/>>.
- [2] *Clipping (zvuk)*. Wikipedia [online]. [cit. 2018-11-12]. Dostupné z: <[https://cs.wikipedia.org/wiki/Clipping_\(zvuk\)](https://cs.wikipedia.org/wiki/Clipping_(zvuk))>.
- [3] COREY, J.; BENSON D. *Audio production and critical listening: technical ear training. Second edition*. New York: Routledge, Taylor & Francis Group, 2017. Audio Engineering Society presents. ISBN 978-1138845947.
- [4] DAILEY, D. *Electronics for guitarists. 2nd ed*. New York: Springer, 2013. s. 141. ISBN 978-1-4614-4086-4.
- [5] DUNCAN, B. *High perfomance audio power amplifiers for music performance and reproduction*. Boston: Newnes, 1996. s. 79. ISBN 978-0750626293.
- [6] *Clipping (photography)*. Wikipedia [online]. [cit. 2018-11-12]. Dostupné z: <[https://en.wikipedia.org/wiki/Clipping_\(photography\)](https://en.wikipedia.org/wiki/Clipping_(photography))>.
- [7] SMITH, J. *Soft Clipping*. Julius Orion Smith III [online]. Stanford, Kalifornie: Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, 2010 [cit. 2018-11-12]. Dostupné z: <https://ccrma.stanford.edu/~jos/pasp/Soft_Clipping.html>.
- [8] ESQUEDA, F.; VÄLIMÄKI V.; BILBAO S. *Aliasing Reduction in Soft-clip Algorithms* [online]. Espoo, Finsko, 2015. Aalto University School of Electrical Engineering, Dept. of Signal Processing and Acoustics. [cit. 2018-11-12]. Dostupné z: <<https://www.eurasip.org/Proceedings/Eusipco/Eusipco2015/papers/1570104119.pdf>>.
- [9] *Using a spectrogram to identify audio problems*. Izotope [online]. 2014 [cit. 2018-11-12]. Dostupné z: <<https://www.izotope.com/en/blog/mastering/using-a-spectrogram-to-identify-audio-problems.html>>.
- [10] *A Complete Guide To Distortion In Electronic Music*. Soundontime [online]. 2018 [cit. 2018-11-12]. Dostupné z: <<https://www.soundontime.com/distortion-electronic-music/>>.
- [11] ZÖLZER, U.; AMATRIAIN X. *DAFX: digital audio effects*. New York, N.Y.: Wiley, 2002. chpt. 5.3.3. ISBN 0-471-49078-4.

- [12] *Hardcore History*. Corehistory [online]. 2009 [cit. 2018-11-12]. Dostupné z: <<http://corehistory.blogspot.com/2009/12/gabber.html>>.
- [13] *Industrial music*. Wikipedia [online]. [cit. 2018-11-12]. Dostupné z: <https://en.wikipedia.org/wiki/Industrial_music>.
- [14] ROBIN, M. *Analog audio noise*. Broadcast Engineering. 2005, Vol. 47, s. 20–26. ISSN 0007-1994.
- [15] MACHAT, K. *Declicking*. Mister's Mastering House [online]. [cit. 2018-11-12]. Dostupné z: <<http://www.about-audio-mastering-software.com/declicking.html>>.
- [16] MAKHOUL J. *Linear prediction: A tutorial review*. IEEE [online]. vol. 63, no. 4, s. 561-580, 1975. [cit. 2018-11-18]. Dostupné z: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1451722>>.
- [17] ADLER, A.; EMIYA, V.; JAFARI, M. G.; ELAD, M.; GRIBONVAL, R.; PLUMBLEY, M. D. *A constrained matching pursuit approach to audio declipping*. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2011, s. 329–332. [cit. 2018-11-18]. Dostupné z: <<http://ieeexplore.ieee.org/document/5946407/>>.
- [18] KITIĆ, S.; JACQUES, L.; MADHU, N.; HOPWOOD, M. P.; SPRIET, A.; DE VLEESCHOUWER, C. *Consistent Iterative Hard Thresholding for Signal Declipping*. 2013. [cit. 2018-11-18]. Dostupné z: <https://www.researchgate.net/publication/235772084_Consistent_Iterative_Hard_Thresholding_For_Signal_Declipping>.
- [19] KITIĆ, S.; BERTIN, N.; GRIBONVAL, R. *Audio Declipping by Cosparsity Hard Thresholding*. iTwist – Traveling Workshop on Interactions between Sparse models and Technology. 2014. [cit. 2018-11-18]. Dostupné z: <<https://hal.inria.fr/hal-00922497>>.
- [20] KITIĆ, S.; BERTIN, N.; GRIBONVAL, R. *Sparsity and Cosparsity for Audio Declipping: A Flexible Non-convex Approach*. LVA/ICA 2015, LNCS 9237 s. 243–250, 2015. [cit. 2018-11-12]. Dostupné z: <<https://arxiv.org/pdf/1506.01830v2.pdf>>.

- [21] ZÁVIŠKA, P.; RAJMIC, P.; PRŮŠA, Z.; VESELÝ, V. *Revisiting synthesis model of Sparse Audio Declipper*. In: LVA ICA 2018 [online]. United Kingdom: Springer, Cham, 2018, 2018, s. 429-445 [cit. 2018-12-05]. DOI: 10.1007/978-3-319-93764-9. Dostupné z: <<https://link.springer.com/content/pdf/10.1007%2F978-3-319-93764-9.pdf>>.
- [22] ZÁVIŠKA, P.; MOKRÝ, O.; RAJMIC, P. *S-SPADE Done Right: Detailed Study of the Sparse Audio Declipper Algorithms*. [online]. 2018 [cit. 2018-12-05]. Dostupné z: <<https://arxiv.org/abs/1809.09847>>.
- [23] ZÁVIŠKA, P.; RAJMIC, P.; MOKRÝ, O.; PRŮŠA, Z. *A Proper version of Synthesis-based Sparse Audio Declipper*. [online]. 2018 [cit. 2018-12-05]. Dostupné z: <<https://ieeexplore.ieee.org/document/8682348>>.
- [24] CANDLES, E. J.; WAKIN, M. B. *An introduction to compressive sampling*. IEEE Signal Processing Magazine, vol. 25, no. 2, 2008: s. 21-30, ISSN 1053-5888.
- [25] ELAD, M. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010, ISBN 9781441970107.
- [26] ADLER, A.; EMIYA, V.; JAFARI, M. G.; ELAD, M.; GRIBONVAL, R.; PLUMBLEY, M. D. *A constrained matching pursuit approach to audio declipping*. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2011, s. 329-332. [cit. 2018-11-12]. Dostupné z: <<http://ieeexplore.ieee.org/document/5946407/>>.
- [27] RAJMIC, P. *Řídké a nízkohodnostní reprezentace signálů s aplikacemi*. Brno, 2014. Habilitační práce. VUT v Brně.
- [28] TSENG, Ch. *Sparse Vectors* [online]. 1999. [cit. 2018-11-12]. Dostupné z: <<https://www.cs.umd.edu/Outreach/hsContest99/questions/node3.html>>.
- [29] DEFRAENE, B.; MANSOUR, N.; HERTOOGH, S. D.; WATERSCHOOOT, T. VAN.; DIEHL, M.; MOONEN, M. *Declipping of Audio Signals Using Perceptual Compressed Sensing*. IEEE Transactions on Audio, Speech, and Language Processing [Online]. 2013, vol. 21. [cit. 2018-11-12]. Dostupné z: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6600777>>.
- [30] HRBÁČEK, R.; RAJMIC, P.; VESELÝ, V.; ŠPIŘÍK, J. *Řídká reprezentace signálů: úvod do problematiky*. Elektrorevue [online]. 2011. ISSN 1213-1539. [cit. 2018-11-12]. Dostupné z: <<http://www.elektrorevue.cz/cz/download/ridke-reprezentace-signalu--uvod-do-problematiky/.e.cz/>>.

- [31] RUDIN, W. *Analýza v reálném a komplexním oboru*. Vyd. 2., přeprac. Praha: Academia, 2003. ISBN 80-200-1125-0.
- [32] ŠPIŘÍK, J.; RAJMIC P.; VESELÝ V. *Reprezentace signálů: od báží k framům*. Elektrorevue [online]. 2010, vol. 12, no. 6, ISSN 1213-1539. [cit. 2018-11-18]. Dostupné z: <<http://www.elektrorevue.cz/cz/clanky/zpracovani-signalu/0/reprezentace-signalu--od-bazi-k-framum/>>.
- [33] ELAD, M. *Sparse and redundant representations: from theory to applications in signal and image processing*. New York: Springer, 2010. Mathematics. ISBN 978-1-4419-7011-4.
- [34] CHRISTENSEN, O. *An introduction to frames and Riesz bases*. Boston: Birkhäuser, 2002. ISBN 0-8176-4295-1.
- [35] CHRISTENSEN, O. *Frames and bases: an introductory course*. Boston: Birkhäuser, 2008. Applied and numerical harmonic analysis. ISBN 978-0-8176-4678-3.
- [36] COMBETTES, P. L.; PESQUET J. C. *Proximal Splitting Methods in Signal Processing*. [online]. 2010 [cit. 2018-12-07]. Dostupné z: <https://www.researchgate.net/publication/45890853_Proximal_Splitting_Methods_in_Signal_Processing>.
- [37] CONDAT, L. *A Generic Proximal Algorithm for Convex Optimization - Application to Total Variation Minimization*. IEEE Signal Processing Letters [online]. IEEE, 2014, roč. 21, s. 985–989. [cit. 2018-12-07]. DOI: 10.1109/LSP.2014.2322123. ISSN 1558-2361. Dostupné z: <<https://ieeexplore.ieee.org/document/6810809>>.
- [38] COMBETTES, P. L.; PESQUET, J. C. *A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery*. [online]. IEEE, 2007, roč. 1, s. 564–574. [cit. 2018-12-08]. DOI: 10.1109/JSTSP.2007.910264. ISSN 1941-0484. Dostupné z: <<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4407760>>.
- [39] CHAMBOLLE, A.; POCK T. *A first-order primal-dual algorithm for convex problems with applications to imaging*. [online]. Springer US, 2010 [cit. 2018-12-08]. ISSN 1573-7683. Dostupné z: <<https://hal.archives-ouvertes.fr/hal-00490826/document>>.

- [40] CONDAT, L. *A Primal–Dual Splitting Method for Convex Optimization Involving Lipschitzian, Proxiable and Linear Composite Terms*. [online]. Springer US, 2012 [cit. 2018-12-08]. DOI: 10.1007/s10957-012-0245-9. ISSN 1573-2878. Dostupné z: <<https://link.springer.com/article/10.1007/s10957-012-0245-9>>.
- [41] SYROVÝ, V. *Hudební akustika. 3., dopl. vyd.*. V Praze: Akademie múzických umění, 2013. Akustická knihovna Zvukového studia Hudební fakulty AMU, s. 57–59. ISBN 978-80-7331-297-8.
- [42] SCHIMMEL, J. *Elektroakustika: skripta k předmětu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2014. ISBN 978-80-214-4716-5.
- [43] FASTL, H.; ZWICKER E. *Psychoacoustics: facts and models. 3rd. ed.*. New York: Springer, 2007. s. 78–83. ISBN 978-3-540-23159-2.
- [44] JOHNSON, D. H. *Signal-to-noise ratio*. Scholarpedia [online]. 2006 [cit. 2018-12-08]. Dostupné z: <http://www.scholarpedia.org/article/Signal-to-noise_ratio>.
- [45] DEFRAENE, B.; VAN WATERSCHOOT T.; FERREAU H. J.; DIEHL M.; MOONEN M. *Real-Time Perception-Based Clipping of Audio Signals Using Convex Optimization*. IEEE [online]. 2012, 20, s. 2657–2671 [cit. 2018-12-09]. DOI: 10.1109/TASL.2012.2210875. ISSN 1558-7924. Dostupné z: <<https://ieeexplore.ieee.org/document/6255767>>.
- [46] ŠPETA, M. *Programová implementace subjektivních testů zvukové kvality*. Brno, 2011. Diplomová práce. VUT v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav Telekomunikací. Vedoucí práce Ing. Ondřej Rášo. Dostupné z URL: <https://dspace.vutbr.cz/xmlui/bitstream/handle/11012/6139/xspeta00_dip_2011.pdf>.
- [47] HUBER, R.; KOLLMEIER B. *PEMO-Q – A new Method for Objective Audio Quality Assessment using a Model of Auditory Perception*. IEEE Transactions on Audio, Speech and Language processing [online]. 2006, roč. 14, s. 1902–1911 [cit. 2019-04-24]. Dostupné z: <<https://ieeexplore.ieee.org/document/1709880>>.
- [48] SØNDERGAARD P. L.; TOREÉÉSANI B.; BALAZS P. *The Linear Time-Frequency Analysis Toolbox*. International Journal of Wavelets, Multiresolution Analysis and Information Processing, 10(4), 2012. Dostupné z URL: <<http://ltfat.github.io/>>.

- [49] *CVX: Matlab Software for Disciplined Convex Programming*, version 2.0 [online]. 2012 [cit. 2018-11-26]. Dostupné z: <<http://cvxr.com/cvx>>.
- [50] GRANT, M.; BOYD S. *Graph implementations for nonsmooth convex programs*. [online]. Springer-Verlag Limited, 2008 [cit. 2018-11-26]. Dostupné z: <https://web.stanford.edu/~boyd/papers/pdf/graph_dcp.pdf>.
- [51] *EBU SQAM CD. EBU Technology & Innovation*. [online]. Ženeva, 2008 [cit. 2019-05-08]. Dostupné z: <<https://tech.ebu.ch/publications/sqamcd>>.
- [52] ZÁVIŠKA, P.; RAJMIC P.; SCHIMMEL J. *Psychoacoustically Motivated Decoding Based on Weighted l_1 Minimization*. [online]. Vysoké učení technické v Brně, Česká Republika, 2019 [cit. 2019-05-09]. Dostupné z: <<https://arxiv.org/abs/1905.00628>>.

Seznam symbolů, veličin a zkratk

ADMM	Alternating Direction Method of Multipliers
CVX	název toolboxu, akronym slova konvexní
DCT	Discrete Cosine transform - diskrétní kosinová transformace
DFT	Discrete Fourier transform - diskrétní Fourierova transformace
DGT	Discrete Gabor transform - diskrétní Gaborova transformace
DI	Distortion index - index zkreslení
ENF	Equal-norm frames - framy s prvky, které mají shodnou normu
ERB	Equivalent Rectangular Bandwidth - ekvivalentní šířka kmitočtového pásma
FT	Fourier transform - Fourierova transformace
GMT	Global masking treshold - globální maskovací práh
GUI	Graphic user interface - uživatelské grafické rozhraní
LASSO	Least Absolute Shrinkage and Selection Operator
LTFAT	Lage Time-Frequency Analysis Toolbox - název Matlab toolboxu pro časově-kmitočtovou analýzu a syntézu dlouhých signálů
MPEG	Moving picture experts group - skupiny vyvíjející standardy používané na kódování audiovizuálních informací
MUSHRA	Multi Stimulus test with Hidden Reference and Anchor - více-stimulový test se skrytou referencí a kotvou
NSP	Null Space Property - vlastnost nulového prostoru
ODG	Objective Difference Grade - rozdílová objektivní stupnice
PA	psychoakustický (model)
PEMO-Q	Perception Model – Quality Assessment - Objektivní metoda měření kvality zvukového signálu
PTF	Parseval tight frame - Parsevalovsky těsný frame
RIP	Restricted Isometry Property - vlastnost zeslabené isometrie
RR	Reliable replace - nahrazení vzorků na spolehlivých pozicích
SDR	Signal to distortion ratio - poměr úrovně signálu k úrovni zkreslení
SNR	Signal to noise ratio - poměr úrovně signálu k úrovni šumu
SPADE	Sparse Audio Declipper - Declipper založený na řídké reprezentaci signálu
SPL	Sound pressure level - hladina akustické tlaku
STFT	Short-Time Fourier transform - krátkodobá Fourierova transformace
UNF	Unit-norm frames - unitární framy
VP	vektorový prostor

Seznam příloh

A Obsah přiloženého CD

85

A Obsah přiloženého CD

Na přiloženém CD jsou k nalezení soubory obsahující tabulky s výsledky testování. Dále také kompletní zdrojový kód algoritmů a funkcí. Ten byl implementován v Matlabu 2016a s použitím toolboxu LTFAT, který je k nalezení zde [48].

Složka s rekonstruovanými signály nazvaná „Rekonstruované zvukové ukázky“ je rozdělena tak, že obsahuje deset složek s odpovídajícími testovacími signály. V každé z těchto deseti složek se nachází dalších pět složek, které odpovídají jednotlivým algoritmům, pomocí kterých byly ukázky rekonstruovány. Tyto složky již obsahují WAV soubory rozdělené podle úrovně, na které došlo ke clippingu. Název souboru obsahuje také hodnoty parametrů algoritmu, se kterými bylo daného výsledku dosaženo.

```
/ ..... kořenový adresář přiloženého CD
├── Výsledky testování
│   ├── výsledky poslechového testu MUSHRA.xlsx
│   └── výsledky testování algoritmů.xlsx
├── Zdrojový kód ..... m-soubory Matlabu
│   ├── audio_samples ..... složka s testovacími soubory
│   │   ├── 01violin.wav
│   │   ├── 02clarinet.wav
│   │   ├── 03bassoon.wav
│   │   ├── 04harp.wav
│   │   ├── 05glockenspiel.wav
│   │   ├── 06celesta.wav
│   │   ├── 07accordion.wav
│   │   ├── 08guitar_sarasate.wav
│   │   ├── 09piano_schubert.wav
│   │   └── 10wind_ensemble.wav
│   ├── condat.m
│   ├── condat_segment.m
│   ├── hard_clip.m
│   ├── main_basic.m ..... soubor pro spuštění výchozího Condatova algoritmu
│   ├── main_psyac.m ..... soubor pro spuštění Condatova algoritmu s PA modelem
│   ├── main_psyac_segment.m .. soubor pro spuštění algoritmu se segmentací signálu
│   ├── masking.m
│   ├── projection.m
│   ├── projection_segment.m
│   ├── reliable_replace.m
│   ├── SDR_calc.m
│   ├── visualization_spectrogram.m
│   ├── visualization_waveform.m
│   ├── wav_generator.m
│   └── weighting_init.m
└── Rekonstruované zvukové ukázky ..... soubory ve formátu WAV
```